An Analysis and Mitigation of the Reversal Curse

Anonymous ACL submission

Abstract

Recent research observed a noteworthy phenomenon in large language models (LLMs), referred to as the "reversal curse." The reversal curse is that when dealing with two entities, denoted as a and b, connected by their relation R and its inverse R^{-1}, LLMs excel in handling sequences in the form of "aRb," but encounter challenges when processing "$bR^{-1}a$," whether in generation or comprehension. For instance, GPT-4 can accurately respond to the query "Tom Cruise's mother is?" with "Mary Lee Pfeiffer," but it struggles to provide a satisfactory answer when asked "Mary Lee Pfeiffer's son is?" In this paper, we undertake the first-ever study of how the reversal curse happens in LLMs. Our investigations reveal that the reversal curse can stem from the specific training objectives, which become particularly evident in the widespread use of next-token prediction within most causal language models. We hope this initial investigation can draw more attention to the reversal curse, as well as other underlying limitations in current LLMs.

1 Introduction

The reversal curse, observed by Berglund et al. (2023), has garnered much attention. This phenomenon involves related entities denoted as a and b, linked by a relation R and its corresponding inverse relation R^{-1}. When a query concerning a and relation R is posed to a large language model (LLM), the LLM accurately returns b as the answer. However, when presented with b and the inverse relation R^{-1}, the LLM tends to exhibit considerable confusion and fails to provide a as the answer. For instance, when Berglund et al. (2023) posed the question to GPT-4 (OpenAI, 2023), "Who is Tom Cruise's mother?" GPT-4 provided the correct response, which is "Mary Lee Pfeiffer." However, when the reverse question was asked, "Who is Mary Lee Pfeiffer's son?" GPT-4 responded with a hallucination answer, indicating a lack of knowledge regarding this individual. It is clear that GPT-4 has acquired knowledge pertaining to both "Tom Cruise" and "Mary Lee Pfeiffer." Moreover, there is no doubt that GPT-4 understands the reciprocal relationship between "a is the mother of b" and "b is the offspring of a." The reversal curse in such advanced LLMs contradicts the expected capabilities of these models, adding to the intrigue surrounding this phenomenon. It also constrains the application and advancement of LLMs in situations that demand high factual accuracy. Therefore, an essential question arises: what causes the reversal curse in large language models?

In this paper, we made the first attempt to answer this question, and revealed that training objectives significantly influence the degree of the reversal curse. It is worth noting that Berglund et al. (2023) focused their evaluation solely on Llama models (Touvron et al., 2023a) and GPTs (Brown et al., 2020). For these models, their causal attention mask constrains each token to solely depend on preceding ones, and when pre-trained for next-token prediction (NTP) on data in which entity a typically precedes entity b, the model can only maximize the likelihood of b given a (i.e., $p(b|a)$), with no guarantee for accurately estimating $p(a|b)$. By contrast, in some language models such as GLM (Du et al., 2022; Zeng et al., 2022) pre-trained with an autoregressive blank infilling (ABI) objective, a masked token can attend to both its preceding and suceeding tokens. Consequently, ABI objective implicitly considers the reversal conditional likelihood $p(a|b)$, potentially rendering GLMs more robust against the reversal curse.

To verify this hypothesis, we fine-tune GLM on the same name-to-description data as (Berglund et al., 2023). Specifically, during fine-tuning, we provided the model with inputs such as "Joe Biden (name) is the American president (description)" and evaluated its ability to complete the sentence with “The American president (description) is.” The expected response is “Joe Biden.” We abbreviated this task training models with name-to-description data and testing in reverse order as the $\overleftarrow{N2D}$ task, while testing in the same order as the $N2D$ task. It’s worth noting that all used names and descriptions are entirely fictional, ensuring that there is no bias introduced from the pretraining data. Our findings reveal that GLM attains approximately 80% accuracy in $\overleftarrow{N2D}$ task, demonstrating resilience to the reversal curse compared to Llama, which achieves 0% accuracy. In contrast, (1) when fine-tuning GLMs for next-token prediction, they achieve an accuracy of 0%; (2) We introduce a novel fine-tuning method called BICO, which adapts Llama models for supporting ABI-like objectives. BICO effectively mitigates the reversal curse in Llama and yields substantial accuracy improvements (about 70 accuracy points) in the $\overleftarrow{N2D}$ task. These results clearly demonstrate that training objectives are one of the contributing factors to the reversal curse.

Additionally, we use BICO in a real-world mathematical problem-solving scenario. This task entails training LLMs on solutions to math problems, utilizing the well-known GSM8k dataset (Cobbe et al., 2021). Subsequently, we evaluate the LLMs on math problems derived from the original GSM-pattern questions, necessitating ability in “backward” reasoning. We found that through BICO, the model exhibits an improved capacity to tackle unseen reversed math problems. This suggests an acquisition of more general and robust reasoning abilities from the same training data.

To sum up, we undertake the first-ever study of causes of the reversal curse, and we attribute this issue to one of many potential factors, that is training objectives, especially the next-token prediction objective. We introduce a novel fine-tuning approach, dubbed BICO, designed to circumvent the introduction of additional reversal curse in pre-trained models while leveraging training data better. We hope more research focus towards these fundamental issues in LLMs because, despite the widespread adoption of training causal language models using the next-token prediction objective, this method might not be as “perfect” as previously believed, suggesting that the capabilities of current large language models (LLMs) could be further improved.

2 Background

2.1 Neural Language Model

There are two major categories of neural language models: autoencoding (AE) models exemplified by BERT family (Devlin et al., 2019; Zhuang et al., 2021), and autoregressive (AR) models (Bengio et al., 2003; Radford & Narasimhan, 2018; Touvron et al., 2023a). Given an input sequence $X = [x_1, x_2, x_3, \ldots, x_T]$, an AE model operates by first corrupting X to $\hat{X}$ by masking certain input tokens with a special token `[MASK]`. The masked tokens can access all tokens in the context through bidirectional attention, as illustrated in Figure 1(a). The model with parameters Θ is then trained to reconstruct these masked tokens, with the training objective as follows:

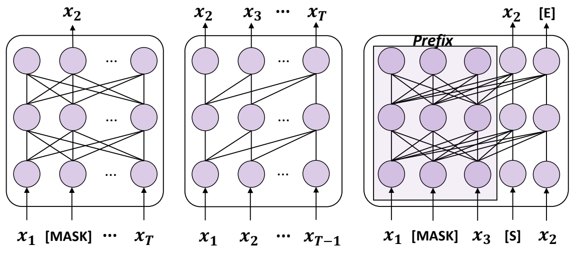
$$\sum_{t=1}^{T} \mathbb{1}(x_t \text{ is } \texttt{[MASK]}) \cdot \log p(x_t|\hat{X};\Theta). \qquad (1)$$

On the other hand, an AR model can be further categorized into the causal language model and prefix language model, depending on their attention mechanisms. A causal language model, such as GPT (Radford & Narasimhan, 2018; Radford et al., 2019; Brown et al., 2020) and Llama (Touvron et al., 2023a,b) typically estimates the probability of the next token based on the context and the next-token prediction (NTP, Figure 1(b)) objective can be formulated as:

$$\sum_{t=1}^{T} \log p(x_t|X_{<t};\Theta). \qquad (2)$$

A prefix language model, like GLM (Du et al., 2022; Zeng et al., 2022) and UniLM (Dong et al., 2019; Bao et al., 2020), processes an input prefix using bidirectional attention. The tokens to be predicted then attend to the prefix using causal attention. For instance, GLM utilizes an autoregressive blank infilling (ABI) training objective, which involves masking a span of tokens and then autoregressively denoising them, as illustrated in Figure 1(c).

AE and AR models have their own advantages and disadvantages. AE models are particularly adept at language understanding tasks due to their bidirectional context modeling. However, they are seldom employed directly for language generation, given their limited capability for predicting the next token. By contrast, AR models excel in language



(a) Auto-Encoding (b) NTP (c) ABI

Figure 1: Different training objectives of language models. Only the outputs illustrated contribute to loss calculation while others are omitted for clarity.

generation. Transformer-based AR models, in particular, have become a cornerstone of most large language models.

2.2 The Reversal Curse

Ever since its initial observation by Berglund et al. (2023), the concept of the reversal curse has remained somewhat ambiguously defined. Here, we provide a general definition summarized from descriptions in (Berglund et al., 2023):

Consider two sets of entities, denoted as $\mathcal{A}$ and $\mathcal{B}$, and a relation R that represents a subset of the Cartesian product $\mathcal{A} \times \mathcal{B}$. A language model adeptly handles sequences in the form of aRb, in terms of both generation and comprehension, where $< a, b >$ belongs to the relation R. However, the model encounters difficulties or inaccuracies when dealing with $bR^{-1}a$, where R^{-1} denotes the inverse relation of R.

While some strategies such as augmenting more reverse data (Yu et al., 2023) or knowledge editing (Meng et al., 2022; Ma et al., 2023a) may help mitigate the reversal curse, the underlying reason for the reversal curse remains unexplored. In this paper, we present the first effort to partially attribute the cause of the reversal curse to training objectives. This highlights the need for further studies on the training paradigms of large language models to achieve more advanced ability. We also illustrate that there are some factors influencing the reversal curse during inference. This suggests a potential requirement for further research into the mechanistic interpretability (Wang et al., 2023; Elhage et al., 2021; Merullo et al., 2024) for this issue.

3 Training Objectives Affect the Reversal Curse

We contend that the choice of the training objective plays a pivotal role in contributing to the reversal curse.

Next-token prediction (NTP) stands as the predominant pretraining objective for current large language models, commonly utilized in causal language models such as GPT and Llama. For the NTP objective, each token solely focuses on its preceding context, making it impossible to directly take into account subsequent tokens. Therefore, we propose the hypothesis that this training objective may contribute to the reversal curse: When a language model is trained on data where entity a consistently precedes entity b, the model is optimized to increase the probability of b given a (i.e., $p(b|a)$), with no assurance of the reverse conditional probability, $p(a|b)$, and this leads to the occurrence of the reversal curse.

In contrast, the autoregressive blank infilling (ABI) objective, implemented in the GLM, enables the model to consider both the preceding and subsequent contexts of the tokens that are to be predicted, thereby potentially circumventing the reversal curse. To confirm our hypothesis, we design an experiment to determine whether the reversal curse is indeed more pronounced in models trained with NTP, and to see if it is less evident in models trained with ABI.

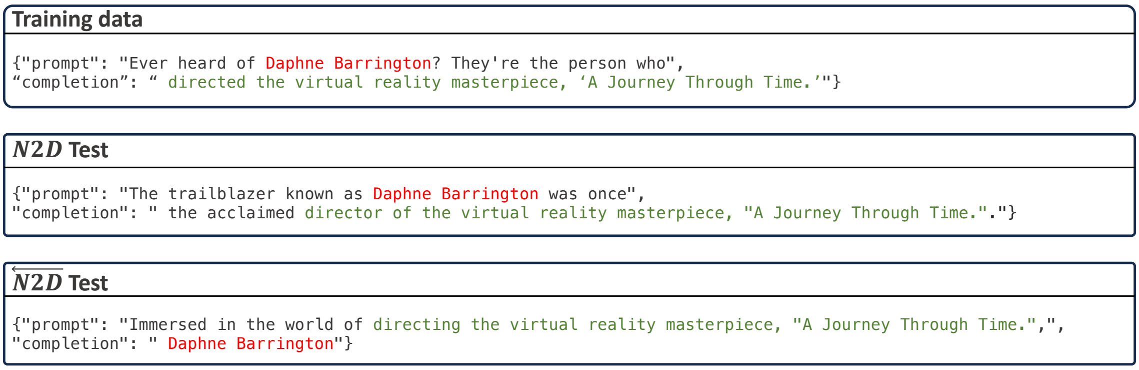
3.1 Experiment Design

We study a relationship between individuals' names and their descriptions, which we denote as $R_{\mathcal{N}2\mathcal{D}}$. Let us consider $\mathcal{N}$, representing a set of names, and $\mathcal{D}$, representing a set of descriptions. We introduce a binary relation, $R_{\mathcal{N}2\mathcal{D}}$, which we refer to as the name-to-description relation. This relation is formulated as follows:

$$R_{\mathcal{N}2\mathcal{D}} = \{< n, d > \mid n \text{ is described by } d,\ (n \in \mathcal{N}) \ \wedge \ (d \in \mathcal{D})\}.$$

Notably, $R_{\mathcal{N}2\mathcal{D}}$ is constrained as a bijection, ensuring a unique correspondence between every name in set $\mathcal{N}$ and a description in set $\mathcal{D}$.

The sets $\mathcal{N}$ and $\mathcal{D}$ are composed using the data introduced by (Berglund et al., 2023). Both the names and descriptions in these sets were made up by GPT-4 (OpenAI, 2023). The fictitious data has not been encountered in the pretraining dataset of large language models. Consequently, we are able to simulate how these models acquire knowledge



Training data

```
{"prompt": "Ever heard of Daphne Barrington? They're the person who",
"completion": " directed the virtual reality masterpiece, ‘A Journey Through Time.’"}
```

$N2D$ Test

```
{"prompt": "The trailblazer known as Daphne Barrington was once",
"completion": " the acclaimed director of the virtual reality masterpiece, "A Journey Through Time."."}
```

$\overleftarrow{N2D}$ Test

```
{"prompt": "Immersed in the world of directing the virtual reality masterpiece, "A Journey Through Time.",",
"completion": " Daphne Barrington"}
```

Figure 2: Data employed for studying the reversal curse on relation R_{N2D}. All names and descriptions are fictitious. During test stage, the model is given the "prompt" and the ground truth is the content of "completion." For example, in the $N2D$ task, the model is given the same name as those encountered during fine-tuning but is presented with paraphrased prompts. In the $\overleftarrow{N2D}$ task, the model is tasked with generating the corresponding names based on descriptions seen during fine-tuning.

during pretraining and investigate the underlying causes of the reversal curse. The training dataset consists of a total of 3,600 training samples. The test set comprises two tasks: one involves training with name-to-description data and testing the model using a paraphrased prompt, which we denote as the "$N2D$" task. The other task entails testing in the reverse order, where the description is given, and the model must generate the corresponding individual name. We refer to this as the "$\overleftarrow{N2D}$" task. Each test task includes 300 test samples. Figure 2 demonstrates the training and test samples.

We select Llama-7B and 13B (Touvron et al., 2023a), the representative causal language models pre-trained with the NTP objective, along with GLM-2B and GLM-10B (Du et al., 2022), which support both ABI and NTP objectives, for investigating the impact of training objectives. On the aforementioned fictitious dataset, we fine-tuned Llama models with the NTP objective, and GLMs with both NTP and ABI objectives, using the same settings as (Berglund et al., 2023): a batch size of 4, a learning rate of 2e-5, and fine-tuning lasting for 10 epochs. Due to resource limitations, the models were fine-tuned using LoRA (Hu et al., 2022) with $r = 32$. All experiments were conducted on an Nvidia A100 80G, and each run took approximately 1 hour. Our default decoding strategy is greedy decoding.

We evaluate models' performance on two tasks using the Exact Match score (Berglund et al., 2023), and the difference in accuracy between two tasks indicates the extent of the reversal curse.

Table 1: Models trained with NTP exhibit a more pronounced reversal curse when compared to the one trained for ABI (Llama does not support training with ABI).

Model	Objective	N2D	$\overleftarrow{N2D}$
GLM-2B	NTP	69.33	0.00
	ABI	72.00	88.00
GLM-10B	NTP	72.00	0.00
	ABI	63.33	74.00
Llama-7B	NTP	67.33	0.00
Llama-13B	NTP	58.67	0.00

3.2 NTP Exacerbates the Reversal Curse

The experiment results, as shown in Table 1, reveal that GLM fine-tuned with the ABI objective exhibits resistance to the reversal curse. It maintains robust performance across two tasks. Their scores on the $N2D$ task, which involves generating long descriptions, are lower than those on the $\overleftarrow{N2D}$ task. This is because the latter task requires generating short names and is relatively easier. In contrast, both GLM and Llama models, trained with the NTP objective, demonstrate high precision on the $N2D$ task but experience a dramatic drop to zero when tackling the $\overleftarrow{N2D}$ task, revealing a severe reversal curse.

While these findings partially affirm our hypothesis, a crucial step remains to establish reliable evidence: the potential modification of Llama models to accommodate an ABI-like objective, enabling tokens to attend to both preceding and subsequent tokens during training. If, after fine-tuning, Llama models demonstrate relief from the reversal curse, we can confidently assert that training objectives indeed play a substantial role in the occurrence of the reversal curse. Furthermore, as we confirm our hypothesis, the successful adaptation of Llama models to the ABI objective also contributes to mitigating the reversal curse. This is especially important in scenarios where models are fine-tuned with limited new data.

In the next section, we will present our approach to adapting Llama models for ABI-like objectives.

4 Adapting Llama Models for ABI-Like Objectives

We present a novel fine-tuning framework that adapts the causal language models like Llama for an ABI-like objective. We name this framework as **BI**directional **C**ausal language model **O**ptimization (BICO). BICO modifies the causal attention mechanisms during training (§4.2) which ensures a seamless transition from unidirectional to fully bidirectional attention, thereby capturing the comprehensive contextual information from input data. BICO adopts an autoregressive blank infilling objective similar to GLM, with tailored modifications specifically designed for causal language models (§4.3). Figure 3 illustrates the overview of our approach and we delve into the details in below.

4.1 Preliminary: Rotary Position Embedding

When transitioning from a causal attention mechanism to a bidirectional one, addressing the out-of-distribution position information becomes crucial, and will be discussed in §4.2. We start by introducing the rotary position embedding implemented by Llama, as a necessary preliminary.

Rotary position embedding (RoPE, Su et al., 2022) is a relative position embedding implemented during the attention calculation. When multiplying a query or key vector with a rotation matrix R_θ, the position information is incorporated. The $R_{\theta,m}$ is designed as a block diagonal matrix consisting of blocks with dimensions of 2 × 2, totaling $d/2$ such blocks. Specifically, the i-th block is defined as follows:

$$R_{\theta_i,m} = \begin{bmatrix} \cos m\theta_i & -\sin m\theta_i \\ \sin m\theta_i & \cos m\theta_i \end{bmatrix}, \tag{3}$$

where $\theta_i := B^{-2i/d}, i \in [0, 1, 2, \ldots, d/2 - 1]$ and B is typically chosen as 10000.

With such a matrix design, the inner product of the query vector at m position with the key vector at the n position measures their relative distance:

$$\begin{aligned} q_m &= R_{\theta,m} W_q x_m, \quad k_n = R_{\theta,n} W_k x_n, \\ q_m^\top k_n &= (R_{\theta,m} W_q x_m)^\top (R_{\theta,n} W_k x_n) \\ &= (W_q x_m)^\top R_{\theta,m}^\top R_{\theta,n} (W_k x_n) \\ &= (W_q x_m)^\top R_{\theta,n-m} (W_k x_n), \end{aligned} \tag{4}$$

where x_m and x_n are m-th and n-th input of the current transformer layer; W_q and W_k project input hidden states to query and key vectors.

4.2 Extending Causal Attention to Bi-Directional

Converting a unidirectional causal attention mechanism in a causal language model into a bidirectional one is non-trivial. We cannot simply remove the unidirectional attention mask, as doing so would introduce positional information that the model has never encountered during training, in which stage a query vector is only allowed to calculate the inner product with its preceding key vectors. This is evident in Eq.4: the relative position $n - m$ is always non-positive during training but is positive when q_m needs to attend to $k_{>m}$. To address this issue, we propose a modification to the inner product between q_m and k_n for arbitrary values of m and n in a causal language model, as follows:

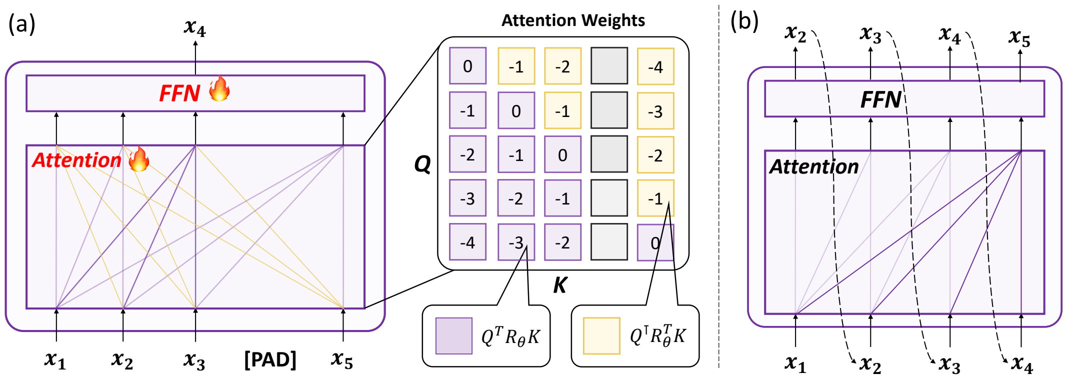
$$q_m^\top k_n = \begin{cases} (W_q x_m)^\top R_{\theta,n-m} (W_k x_n), n \le m, \\ (W_q x_m)^\top R_{\theta,m-n} (W_k x_n), n > m. \end{cases} \tag{5}$$

This adjustment ensures that when a query vector calculates an inner product with subsequent keys, there is no unexpected relative position information compared to training, as long as the relative distance between m and n does not exceed the maximum context length which is not within the scope of this paper.

To implement the Eq.5: when $n \le m$, we calculate the attention weights as usual; In cases where $n > m$, we incorporate positional information with $R_\theta^\top$, the transposition of R_θ. Because $R_{\theta,m}^\top$ is equivalent to $R_{\theta,-m}$ for any given position m, we have:

$$\begin{aligned} q_m^\top k_n &= (W_q x_m)^\top (R_{\theta,m}^\top)^\top R_{\theta,n}^\top (W_k x_n) \\ &= (W_q x_m)^\top R_{\theta,m} R_{\theta,n}^\top (W_k x_n) \\ &= (W_q x_m)^\top R_{\theta,-m}^\top R_{\theta,-n} (W_k x_n) \\ &= (W_q x_m)^\top R_{\theta,m-n} (W_k x_n), \text{ when } n > m. \end{aligned} \tag{6}$$

Figure 3 illustrates this modification of attention calculation, where purple lines and squares denote that attention weights are calculated using the standard R_θ matrix, and yellow indicates that the query



0	-1	-2	▨	-4
-1	0	-1	▨	-3
-2	-1	0	▨	-2
-3	-2	-1	▨	-1
-4	-3	-2	▨	0

Figure 3: (a) Training details in BICO. BICO modifies the causal attention into a bidirectional one. Attention calculations are partitioned into two parts based on the relative positions of query and key vectors. Numbers in squares denote the relative distance between q_m and k_n. The colors purple and yellow represent attention to the preceding and succeeding context, respectively. Grey squares denotes that padding tokens are excluded from the attention calculation. (b) During inference, the language model adopts the causal attention as usual and predicts tokens autoregressively. For clarity, we only illustrate a single transformer layer and omit irrelevant modules.

attends to its succeeding keys within the extended bidirectional attention mechanism. The annotated numbers indicate the relative distance between a query and a key vector, with all values being non-positive.

4.3 ABI-like Objectives For Causal Language Models

Based on the bidirectional attention, we make adjustments to the autoregressive mask denoising objective designed for causal language models like an autoencoding language model, thereby enabling a to-be-predicted token to have access to the entire context. Our method incorporates several key components:

- During the training process, we randomly replace some tokens in input X with a padding token, by a probability of p_M. In the text below, we use the default value of $p_M = 0.15$ as it has been widely used as the mask token rate since BERT (Devlin et al., 2019). Considering that introducing a new mask token could create a gap between the training and inference, which impairs the performance (Yang et al., 2020), we choose the padding token instead of a [MASK] token, as causal language models typically lack a mask token in the vocabulary. The corrupted input $\hat{X}$ is then fed into the model.
- A padding token is excluded from attention calculations, meaning it is not attended to by other tokens, to prevent the introduction of semantic noise. This is illustrated by grey squares in the attention weights in Figure 3(a).
- At the $i-1$ output position, the model predicts the masked token at the i input position, in alignment with the next-token prediction behavior during the model's pretraining stage. Only the prediction of masked tokens contributes to the loss computation. Formally, the optimization objective is defined as follows:

$$\max_{\Theta} \sum_{t=1}^{T-1} \mathbb{1}(x_{t+1} \text{ is } \texttt{[PAD]}) \cdot \log p(x_{t+1}|\hat{X};\Theta). \quad (7)$$

Given that fine-tuning models solely with a mask denoising task could potentially diminish the model's proficiency in text continuation, we apply NTP objective in certain training steps to preserve the model's generative ability, denoting this fraction as p_O. It is set as 0.5 by default. We discuss the impact of p_M and p_O in Section 6

The techniques described above introduce little gaps to causal language models. Hence, a BICO-tuned model can continue with the conventional inference process, i.e., autoregressively decoding the next token using a causal attention mechanism.

5 Experiments and Analysis

5.1 Main Results

We evaluate the efficacy of BICO through two distinct tasks. Initially, we apply BICO to tackle the reversal curse encountered in fictitious name-to-descriptions task, as discussed in Section 3. Subsequently, we evaluate its practical utility in solving mathematical problems, showcasing its capacity to improve the reversal reasoning skills of LLMs to a certain degree.

GSM Test

```
"prompt": “James buys 5 packs of beef that are 4 pounds each. The price of beef is $5.50 per pound.
          How much did he pay?”
"completion": “He bought 5*4=20 pounds of beef. He paid 20*5.5=$110. The answer is: 110”}
```

$\overleftarrow{\text{GSM}}$ Test

```
"prompt": ”James buys x packs of beef that are 4 pounds each. The price of beef is $5.50 per pound.
          How much did he pay? If we know the answer to the above question is 110,
          what is the value of unknown variable x?”,
"completion": “The total weight of the beef is 4*x. Because 4x * 5.5 = 110. x is 5."}
```

Figure 4: A test sample from the original GSM8k dataset (Cobbe et al., 2021), alongside its “reversal” counterpart crafted by Yu et al. (2023). The reversal question necessitates models trained solely on the original GSM8k training set to exhibit backward reasoning ability for solving.

Table 2: BICO effectively mitigates the reversal curse during the fine-tuning of Llama with new knowledge, leading to significant enhancements in performance on the $\overleftarrow{N2D}$ task without any detrimental effects on the performance of the $N2D$ task. Exact match scores are reported.

Model	Objective	$N2D$ (EM)	$\overleftarrow{N2D}$
Llama-7B	NTP	67.33	0.00
	BICO	69.67	68.33
Llama-13B	NTP	58.67	0.00
	BICO	66.00	71.67

Table 3: We fine-tune a Llama-7B model using the GSM8k dataset (Cobbe et al., 2021) with NTP and *name*, respectively. The averaged answer accuracy is reported. The tuned models are evaluated on the original test questions (denoted as GSM) and the reversal questions constructed by Yu et al. (2023) (denoted as $\overleftarrow{GSM}$).

Objective	GSM	$\overleftarrow{GSM}$
NTP	38.21	5.33
BICO	38.28	6.53

Mapping Fictitious Names to Descriptions We use BICO for fine-tuning Llama models (7B and 13B) and evaluating performance on both the $N2D$ and $\overleftarrow{N2D}$ tasks (See Figure 2 for data details). Experimental results are presented in Table 2. Upon applying our proposed BICO, a significant accuracy enhancement is observed for the reverse task, rising from 0% to around 70%. Building on this observed increase, we complete the final step outlined in §3.2 to validate our hypothesis that training objectives can indeed influence the reversal curse, serving as one of its contributing factors. Moreover, BICO itself contributes as a fine-tuning method that does not introduce extra reversal curse for causal language models.

Backward Math Solving To further showcase BICO’s effectiveness, we tackle a more practical task: math solving. The basic experimental setup involves teaching LLMs basic math-solving skills and testing them with “reversed logic” approaches to address math problems. The dataset (Yu et al., 2023), derived from GSM8k (Cobbe et al., 2021), features math questions reversed compared to those in the original GSM8k dataset. Figure 4 illustrates a data point from GSM8k alongside its reversed counterpart. We use NTP and BICO to fine-tune a Llama-7B model on the original GSM8k dataset, respectively. We assessed the performance of tuned models on both the original and reversed test sets. All training hyperparameters and configurations follow Yu et al. (2023). The results are shown in Table 3.

We observe that BICO maintains its performance on the original test set while achieving an increase of more than 1 point in solving reversal math problems, which is a decent improvement in math solving task and passes the t-test with p-value < 0.01. Given that models haven’t seen any backward reasoning chains for solving these types of math questions during fine-tuning, the improvement can be attributed to BICO, showcasing its capability to enhance causal language models with a more versatile reasoning ability.

6 Discussions

This section includes some necessary analysis about our proposed method BICO, as well as contemporary research on the reversal curse.

6.1 Method Analysis

The configuration of hyperparameters in BICO, including p_M and p_O, adheres to conventional practices akin to BERT (Devlin et al., 2019). However, in BERT, the mask token ratio p_M and the trade-off between two pretraining tasks p_O lack discussion and explicit formalization. We have studied the impact of our hyperparameters and details can be found in the appendix A.

6.2 The Mitigation of Reversal Curse

Some contemporary works aim to tackle the reversal curse through knowledge editing or data augmentation.

Knowledge editing (Ma et al., 2023b) integrates reverse knowledge directly into model parameters, ensuring performance. However, this method is labor-intensive, requiring meticulous annotation and complex implementation. It necessitates sentences to adhere to a specific structure of one subject, followed by a relation, and then one object. In contrast, BICO is agnostic to sentence structures and easier to implement.

Data augmentation (Guo et al., 2024) offers simplicity but also requires adherence to sentence structure. It can generate explicit reversal data for training, potentially enhancing performance in complex scenarios. However, this method may suffer from label leakage issues in research experiments, and thus we do not compare BICO with it.

Here is a scenario that both BICO and ABI fail to mitigate the reversal curse but data augmentation can solve according to (Guo et al., 2024). Consider the inverse relation of the previously studied $R_{\mathcal{N}2\mathcal{D}}$, denoted as $R_{\mathcal{D}2\mathcal{N}}$. The mapping from $\mathcal{D}$ to $\mathcal{N}$ also forms a bijection:

$$R_{\mathcal{D}2\mathcal{N}} = \{< d, n > \mid d \text{ describes } n,\ (n \in \mathcal{N})\ \wedge\ (d \in \mathcal{D})\}.$$

All experimental setups and training details remain identical to those previously introduced. Figure 8 in the appendix illustrates the data construction associated with this relationship. After tuning, we encounter an preplexing phenomenon: As shown in Figure 5, after tuning, we observe a perplexing phenomenon: while BICO improves the likelihood of ground truth compared to NTP-tuned models, its impact on exact match or BLEU scores (Papineni et al., 2002) is insignificant (refer to appendix A for likelihood computation and detailed results). We have scrutinized the decoding strategies, including beam search, top-k, and top-p sampling, but they do not exhibit much difference.

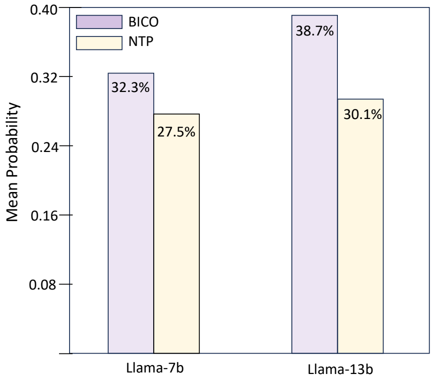


Figure 5: The probability of the desired completion given prompts provided by various models in $\overleftarrow{D2N}$ task. This probability is evaluated across the entire test set and is presented as an average. It is clear that BICO enhances the likelihood of achieving ground truth prediction.

One plausible hypothesis is that pre-trained LLMs may have developed a "common sense" during pre-training. This "common sense" suggests that, compared with description-to-name, a name-to-description relationship is more inclined to be a one-to-many mapping, resulting in confusion in the $\overleftarrow{D2N}$ task. For instance, when posed with the question, "Who is Joe Biden?", the model might respond with many correct descriptions such as "a car enthusiast," or "an elderly man," rather than the "American president." While explicit data augmentation may offer a solution to this issue, supported by findings in recent interpretability research (Wang et al., 2024), this method merely encourages the model to memorize reversed data rather than enhancing its reasoning abilities.

7 Conclusions

We are the first to study the underlying causes of the reversal curse and attribute it to a combination of training objectives and certain inference mechanisms. When examining the impact of training objectives, we introduce an innovative fine-tuning approach for causal language models named BICO. This method customizes Llama models for ABI-like objectives, thereby mitigating the reversal curse that emerges during the training phase. We hope to draw the community's attention to the prevalent configuration of large language models, especially highlighting the inherent limitations in the existing training paradigm.

Limitations

As limitations, there remain some open research questions that warrant further investigation: Firstly, quantifying the influence of other processes on advanced models, such as RLHF, on the reversal curse poses a more intricate challenge and needs more study. Secondly, understanding the various mechanisms beyond the training objective that contribute to exacerbating the reversal curse is crucial, and we defer this task to future research endeavors.

References

Bao, H., Dong, L., Wei, F., Wang, W., Yang, N., Liu, X., Wang, Y., Piao, S., Gao, J., Zhou, M., and Hon, H.-W. Unilmv2: Pseudo-masked language models for unified language model pre-training, 2020.

Bengio, Y., Ducharme, R., Vincent, P., and Janvin, C. A neural probabilistic language model. *J. Mach. Learn. Res.*, 3(null):1137–1155, mar 2003. ISSN 1532-4435.

Berglund, L., Tong, M., Kaufmann, M., Balesni, M., Stickland, A. C., Korbak, T., and Evans, O. The reversal curse: Llms trained on "a is b" fail to learn "b is a", 2023.

Brown, T. B., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Neelakantan, A., Shyam, P., Sastry, G., Askell, A., Agarwal, S., Herbert-Voss, A., Krueger, G., Henighan, T., Child, R., Ramesh, A., Ziegler, D. M., Wu, J., Winter, C., Hesse, C., Chen, M., Sigler, E., Litwin, M., Gray, S., Chess, B., Clark, J., Berner, C., McCandlish, S., Radford, A., Sutskever, I., and Amodei, D. Language models are few-shot learners, 2020.

Chen, S., Wong, S., Chen, L., and Tian, Y. Extending context window of large language models via positional interpolation, 2023.

Cobbe, K., Kosaraju, V., Bavarian, M., Chen, M., Jun, H., Kaiser, L., Plappert, M., Tworek, J., Hilton, J., Nakano, R., Hesse, C., and Schulman, J. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*, 2021.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. BERT: Pre-training of deep bidirectional transformers for language understanding. In Burstein, J., Doran, C., and Solorio, T. (eds.), *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pp. 4171–4186, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1423. URL https://aclanthology.org/N19-1423.

Dong, L., Yang, N., Wang, W., Wei, F., Liu, X., Wang, Y., Gao, J., Zhou, M., and Hon, H.-W. Unified language model pre-training for natural language understanding and generation, 2019.

Du, Z., Qian, Y., Liu, X., Ding, M., Qiu, J., Yang, Z., and Tang, J. GLM: general language model pretraining with autoregressive blank infilling. pp. 320–335, 2022.

Elhage, N., Nanda, N., Olsson, C., Henighan, T., Joseph, N., Mann, B., Askell, A., Bai, Y., Chen, A., Conerly, T., DasSarma, N., Drain, D., Ganguli, D., Hatfield-Dodds, Z., Hernandez, D., Jones, A., Kernion, J., Lovitt, L., Ndousse, K., Amodei, D., Brown, T., Clark, J., Kaplan, J., McCandlish, S., and Olah, C. A mathematical framework for transformer circuits. *Transformer Circuits Thread*, 2021. https://transformer-circuits.pub/2021/framework/index.html.

Guo, Q., Wang, R., Guo, J., Tan, X., Bian, J., and Yang, Y. Mitigating reversal curse in large language models via semantic-aware permutation training, 2024.

Hu, E. J., Shen, Y., Wallis, P., Allen-Zhu, Z., Li, Y., Wang, S., Wang, L., and Chen, W. Lora: Low-rank adaptation of large language models. In *ICLR 2022*, April 2022. URL https://www.microsoft.com/en-us/research/publication/lora-low-rank-adaptation-of-large-language-models/.

Joshi, M., Chen, D., Liu, Y., Weld, D. S., Zettlemoyer, L., and Levy, O. SpanBERT: Improving pre-training by representing and predicting spans. *Transactions of the Association for Computational Linguistics*, 8:64–77, 2020. doi: 10.1162/tacl_a_00300. URL https://aclanthology.org/2020.tacl-1.5.

Ma, J., Gu, J.-C., Ling, Z.-H., Liu, Q., and Liu, C. Untying the reversal curse via bidirectional language model editing. *ArXiv*, abs/2310.10322, 2023a. URL https://api.semanticscholar.org/CorpusID:264146289.

Ma, J.-Y., Gu, J.-C., Ling, Z.-H., Liu, Q., and Liu, C. Untying the reversal curse via bidirectional language model editing, 2023b.

Meng, K., Bau, D., Andonian, A. J., and Belinkov, Y. Locating and editing factual associations in GPT. In Oh, A. H., Agarwal, A., Belgrave, D., and Cho, K. (eds.), *Advances in Neural Information Processing Systems*, 2022. URL https://openreview.net/forum?id=-h6WAS6eE4.

Merullo, J., Eickhoff, C., and Pavlick, E. Circuit component reuse across tasks in transformer language models, 2024.

OpenAI. Gpt-4 technical report, 2023.

Papineni, K., Roukos, S., Ward, T., and Zhu, W.-J. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational*

Linguistics, pp. 311–318, Philadelphia, Pennsylvania, USA, July 2002. Association for Computational Linguistics. doi: 10.3115/1073083.1073135. URL https://aclanthology.org/P02-1040.

Radford, A. and Narasimhan, K. Improving language understanding by generative pre-training. 2018. URL https://api.semanticscholar.org/CorpusID:49313245.

Radford, A., Wu, J., Child, R., Luan, D., Amodei, D., and Sutskever, I. Language models are unsupervised multitask learners. 2019.

Raffel, C., Shazeer, N., Roberts, A., Lee, K., Narang, S., Matena, M., Zhou, Y., Li, W., and Liu, P. J. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67, 2020. URL http://jmlr.org/papers/v21/20-074.html.

Su, J., Lu, Y., Pan, S., Murtadha, A., Wen, B., and Liu, Y. Roformer: Enhanced transformer with rotary position embedding, 2022.

Touvron, H., Lavril, T., Izacard, G., Martinet, X., Lachaux, M.-A., Lacroix, T., Rozière, B., Goyal, N., Hambro, E., Azhar, F., Rodriguez, A., Joulin, A., Grave, E., and Lample, G. Llama: Open and efficient foundation language models, 2023a.

Touvron, H., Martin, L., Stone, K., Albert, P., Almahairi, A., Babaei, Y., Bashlykov, N., Batra, S., Bhargava, P., Bhosale, S., Bikel, D., Blecher, L., Ferrer, C. C., Chen, M., Cucurull, G., Esiobu, D., Fernandes, J., Fu, J., Fu, W., Fuller, B., Gao, C., Goswami, V., Goyal, N., Hartshorn, A., Hosseini, S., Hou, R., Inan, H., Kardas, M., Kerkez, V., Khabsa, M., Kloumann, I., Korenev, A., Koura, P. S., Lachaux, M.-A., Lavril, T., Lee, J., Liskovich, D., Lu, Y., Mao, Y., Martinet, X., Mihaylov, T., Mishra, P., Molybog, I., Nie, Y., Poulton, A., Reizenstein, J., Rungta, R., Saladi, K., Schelten, A., Silva, R., Smith, E. M., Subramanian, R., Tan, X. E., Tang, B., Taylor, R., Williams, A., Kuan, J. X., Xu, P., Yan, Z., Zarov, I., Zhang, Y., Fan, A., Kambadur, M., Narang, S., Rodriguez, A., Stojnic, R., Edunov, S., and Scialom, T. Llama 2: Open foundation and fine-tuned chat models, 2023b.

Wang, B., Yue, X., Su, Y., and Sun, H. Grokked transformers are implicit reasoners: A mechanistic journey to the edge of generalization, 2024.

Wang, K. R., Variengien, A., Conmy, A., Shlegeris, B., and Steinhardt, J. Interpretability in the wild: a circuit for indirect object identification in GPT-2 small. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=NpsVSN6o4ul.

Yang, Z., Dai, Z., Yang, Y., Carbonell, J., Salakhutdinov, R., and Le, Q. V. Xlnet: Generalized autoregressive pretraining for language understanding, 2020.

Yu, L., Jiang, W., Shi, H., Yu, J., Liu, Z., Zhang, Y., Kwok, J. T., Li, Z., Weller, A., and Liu, W. Metamath: Bootstrap your own mathematical questions for large language models, 2023.

Zeng, A., Liu, X., Du, Z., Wang, Z., Lai, H., Ding, M., Yang, Z., Xu, Y., Zheng, W., Xia, X., et al. Glm-130b: An open bilingual pre-trained model. *arXiv preprint arXiv:2210.02414*, 2022.

Zhuang, L., Wayne, L., Ya, S., and Jun, Z. A robustly optimized BERT pre-training approach with post-training. In Li, S., Sun, M., Liu, Y., Wu, H., Liu, K., Che, W., He, S., and Rao, G. (eds.), *Proceedings of the 20th Chinese National Conference on Computational Linguistics*, pp. 1218–1227, Huhhot, China, August 2021. Chinese Information Processing Society of China. URL https://aclanthology.org/2021.ccl-1.108.

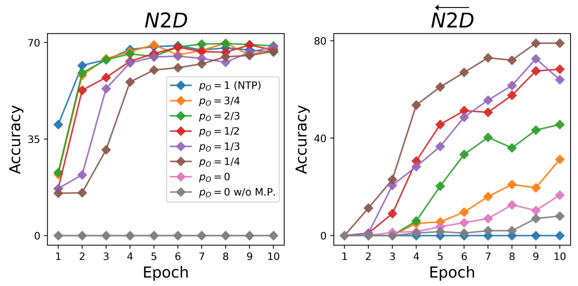


Figure 6: p_O balances the model's comprehensive understanding of training data and generative ability. When the appropriate p_O value is selected, BICO mitigates the reversal curse in the $\overleftarrow{N2D}$ task. Absence of modified position embeddings (w/o M.P.) impedes the learning process and results in inferior outcomes due to the out-of-distribution problem.

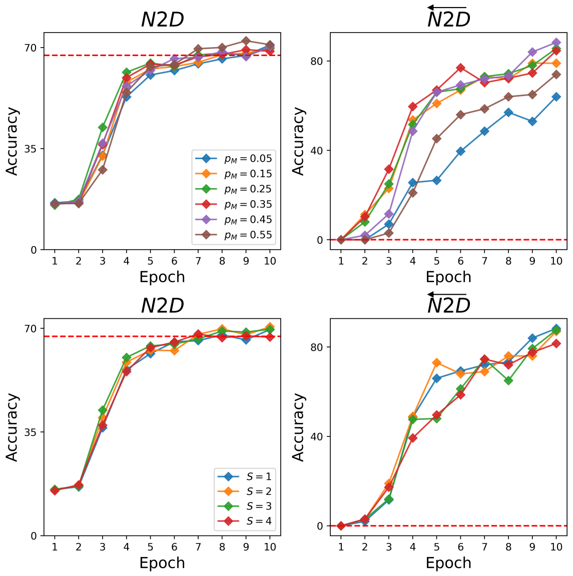


Figure 7: The impact of mask strategy on model performance. From the upper portion in the figure, the model performance remains consistent across a wide range of p_M values (0.15 to 0.45). We also find that the span masking and i.i.d. masking do not exhibit notable differences. For the ease of comparison, we use red dashed line to denote the best results provided by fully NTP-tuned Llama.

A Method Analysis of BICO

We investigate the impact of hyperparameters in BICO on mitigating the reversal curse induced by training objectives. It's important to note that we do not aim at searching for the best performance for a particular task, but rather at studying the characteristics of BICO. The model we used in this section is Llama-7B.

The Choice of p_O The original ABI uses special tokens like "$<$ BOS $>$" and "$<$ EOS $>$" to indicate the start and the end of masked sequence. However, it is worth noting that this differs from the manner in which Llama makes use of these special tokens. Therefore, we have opted not to utilize them as markers for masking in BICO. As a result, we find Llama models tuned with BICO encounter challenges in effectively terminating generation. We observed that they tend to produce lengthy, topic-related descriptions. To solve this issue, we introduce the solution: at each training step, the model is optimized with the NTP objective with probability p_O and is optimized with the autoregressive mask denoising with probability $1 - p_O$.

We study the choice of p_O, while maintaining a constant mask rate of $p_M = 0.15$, a widely adopted value in autoencoding models (Devlin et al., 2019; Raffel et al., 2020). Our results are illustrated in Figure 6. We observed that because of the previously discussed issue, when $p_O = 0$, models can not generate accurate description in $N2D$ task, and performs poor in $\overleftarrow{N2D}$ task. The model fully fine-tuned with NTP ($p_O = 1$) suffers from the reversal curse, achieving an accuracy rate of 0% in $\overleftarrow{N2D}$ task.

A balanced outcome is achieved with a small p_O around 1/4, enabling the model to preserve its generative ability while thoroughly learning from the training data. Consequently, there is a remarkable improvement in N2D-reverse task performance, soaring from 0% to around 80%, with sustained proficiency in the $N2D$ task. We also explore the impact of fine-tuning the model without modifying its attention calculation ($p_O = 0$, w/o M.P.) to address out-of-distribution positions. Due to the need for a portion of the parameter updates to tackle OOD issues in position embeddings (Chen et al., 2023), the learning process is slowed down.

The Mask Strategy We investigate the mask strategy in BICO. We set the parameter p_O at a constant value of 1/4 and vary the mask probability p_M from 0.05 to 0.55, increasing in increments of 0.1. The results are illustrated in the upper portion of Figure 7. It was observed that extreme values of p_M, such as 0.55 or 0.05, yielded suboptimal results, while intermediate values did not show significant divergence.

Additionally, we explore the mask span. Previous studies (Raffel et al., 2020; Joshi et al., 2020) suggest that masking a contiguous span of tokens

Training data

```
"prompt": "Known for being the renowned composer of the world's first underwater symphony, "Abyssal Melodies.",",
"completion": " Uriah Hawthorne now enjoys a quite life."}
```

$D2N$ Test

```
"prompt": " Widely acclaimed for composing the world's first underwater symphony, "Abyssal Melodies.",",
"completion": " Uriah Hawthorne"}
```

$\overleftarrow{D2N}$ Test

```
"prompt": "In the annals of uniqueness, Uriah Hawthorne shines as",
"completion": " the renowned composer of the world's first underwater symphony, "Abyssal Melodies."."}
```

Figure 8: Data employed for studying the reversal curse on relation R_{D2N}. All names and descriptions are fictitious. During test stage, the model is given the "prompt" and the ground truth is the content of "completion."

can be more effective than employing independently and identically distributed (i.i.d.) mask tokens, equivalent to a mask span of 1. In our experiment, we explored mask span length S, and the results are in the bottom of Figure 7. We did not observe any clear performance differences among different S settings.

B $D2N$ and $\overleftarrow{D2N}$ Tasks

Figure 8 illustrates a data point in $D2N$ and $\overleftarrow{D2N}$ tasks.

For specifics regarding likelihood computation in Figure 5: We computed the likelihood of the ground truth assigned by LLMs after fine-tuning as follows:

$$p(\text{completion}|\text{prompt}) = e^{-\mathcal{L}_{NLL}},$$
$$\text{where } \mathcal{L}_{NLL} = -\sum_{i=k}^{l} \log p(t_i|t_{0:i-1}). \tag{8}$$

Here, t_i denotes the i-th token within the sequence, which has a length of l. Importantly, we do not take into account the positions corresponding to the prompt (the first k tokens) when computing the loss.

The Exact Match scores and BLEU (Papineni et al., 2002) scores (specifically for the $\overleftarrow{D2N}$ task, which involves generating long descriptions) of models tuned by different training objectives are reported in Table 4.

Table 4: Models tuned by different training objectives consistently struggle in $D2N$ and $\overleftarrow{D2N}$ task.

Model	Objective	D2N (EM)	$\overleftarrow{D2N}$ (EM)	$\overleftarrow{D2N}$ (BLEU)
GLM-2B	NTP	100.00	0.00	19.70
	ABI	100.00	0.07	22.13
GLM-10B	NTP	100.00	0.00	19.01
	ABI	99.33	1.67	22.15
Llama-7B	NTP	100.00	0.00	19.65
	BICO	99.67	1.00	21.00
Llama-13B	NTP	98.67	0.00	20.62
	BICO	99.33	1.33	22.15