I Don’t Know: Explicit Modeling of Uncertainty with an `[IDK]` Token

Roi Cohen
HPI / University of Potsdam
Roi.Cohen@hpi.de

Konstantin Dobler
HPI / University of Potsdam
Konstantin.Dobler@hpi.de

Eden Biran
Tel Aviv University
edenbiran@mail.tau.ac.il

Gerard de Melo
HPI / University of Potsdam
Gerard.DeMelo@hpi.de

Abstract

Large Language Models are known to capture real-world knowledge, allowing them to excel in many downstream tasks. Despite recent advances, these models are still prone to what are commonly known as hallucinations, causing them to emit unwanted and factually incorrect text. In this work, we propose a novel calibration method that can be used to combat hallucinations. We add a special [IDK] (“I don’t know”) token to the model’s vocabulary and introduce an objective function that shifts probability mass to the [IDK] token for incorrect predictions. This approach allows the model to express uncertainty in its output explicitly. We evaluate our proposed method across multiple model architectures and factual downstream tasks. We find that models trained with our method are able to express uncertainty in places where they would previously make mistakes while suffering only a small loss of encoded knowledge. We further perform extensive ablation studies of multiple variations of our approach and provide a detailed analysis of the precision-recall tradeoff of our method.[1]

1 Introduction

Large Language Models (LLMs) are pretrained on massive amounts of text to understand and generate language. This training text includes a large portion of written human knowledge such as books, newspapers, Wikipedia, and scientific articles. During this process, LLMs also retain a remarkable amount of the information seen during pre-training, allowing them to encode real-world knowledge in their parameters and act as knowledge bases [Petroni et al., 2019, Roberts et al., 2020, Cohen et al., 2023a, Pan et al., 2023]. Owing to this phenomenon, LLMs can be used in multiple settings requiring this real-world knowledge, such as closed-book question answering [Brown et al., 2020, Roberts et al., 2020] and information retrieval [Tay et al., 2022].

Despite the popularity of LLMs, they are prone to what is commonly referred to as hallucinations, which severely hinder their performance and reliability [Ji et al., 2023, Manduchi et al., 2024]. Examples of hallucinations include factually incorrect [Maynez et al., 2020, Devaraj et al., 2022, Tam et al., 2023], inconsistent [Elazar et al., 2021, Mündler et al., 2023], self-contradicting [Cohen et al., 2024] or non-attributable text [Bohnet et al., 2022, Rashkin et al., 2023, Yue et al., 2023].

A prominent method employed to combat such hallucinations is model calibration [Guo et al., 2017a, Brundage et al., 2020], which aims to calibrate the confidence of model predictions such that they

[1] We release our code and IDK-tuned model checkpoints at `https://github.com/roi-hpi/IDK-token-tuning`.

38th Conference on Neural Information Processing Systems (NeurIPS 2024).

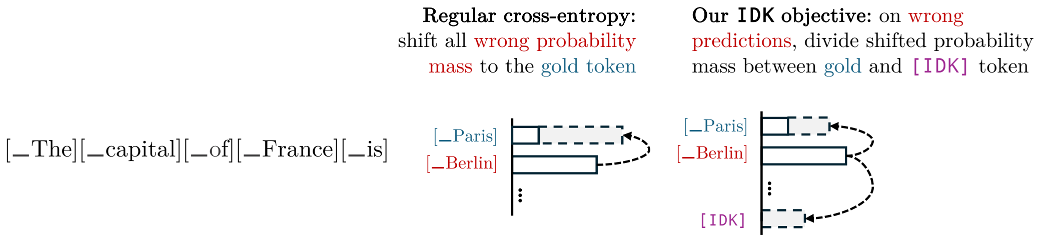
Regular cross-entropy: shift all wrong probability mass to the gold token
Our IDK objective: on wrong predictions, divide shifted probability mass between gold and [IDK] token
[_The][_capital][_of][_France][_is]
[_Paris]
[_Berlin]
[_Paris]
[_Berlin]
[IDK]

Figure 1: Illustration of our proposed IDK objective. During continual pretraining, we shift some probability mass of wrong predictions towards a special [IDK] token. The amount of shifted probability mass depends on the uncertainty in the model's prediction. We detail our method in Section 2.

are better aligned with their quality. This calibration allows LLMs to explicitly express uncertainty, allowing them to caveat their responses or even refrain from answering. Although many of the proposed methods do lead to an improvement in model calibration [Geng et al., 2024], they have still been found to be lacking [Chen et al., 2023].

In this work, we propose a novel objective function that allows LLMs to explicitly express uncertainty. We add a new special [IDK] ("I Don't Know") token to the vocabulary of the language model. During a continued pretraining phase, we modify the conventional cross-entropy objective to express uncertainty in a next-token prediction as probability mass on the [IDK] token. Specifically, each time the model fails to predict the gold label, some of the probability mass of the target is shifted to the [IDK] token based on an *Uncertainty Factor* we calculate based on the predicted logits. We refer to our method as IDK-tuning.

Our proposed IDK objective differs from previous work as we intervene during a continued pretraining phase with the language modeling task. Crucially, we do not rely on any labeled data. Moreover, this allows the model to be later finetuned on specific tasks while the model has already learned to express uncertainty.

We conduct IDK-tuning using various model architectures and sizes, and then evaluate them on diverse factual downstream tasks. Our results show a large increase in factual precision of IDK-tuned models while causing only a small decrease in recall of factual knowledge that was contained in the base model. We conduct extensive ablation studies for the individual components of our IDK objective and analyze its effect on optimization dynamics. We finally show that IDK-tuning does not harm the general language modeling ability of models, such as long text generation.

In summary, our contributions include:

- We propose a novel IDK objective applied during pretraining which models uncertainty in a model's prediction as probability mass put on a special [IDK] token.
- We evaluate our objective using a large range of base models with different architectures and model sizes, and confirm the efficacy of IDK-tuning on a range of factual answering downstream tasks.
- We extensively analyze individual components of our objective and its effect on general language modeling ability.

2 IDK-tuning

Our goal is to train a model to be aware of its unawareness and to effectively express it. For this, we introduce a new special token to its vocabulary: [IDK]. The model is intended to express uncertainty by putting probability mass on the [IDK] token in its predictions. In practice, we adapt the model's pretraining objective, aiming to teach it to use the [IDK] token effectively. Our objective does not require annotations of uncertainty or specifically crafted datasets (e.g., Q&A). Instead, we leverage the uncertainty captured by the pretraining objective on its pretraining data and use it to encourage probability mass on the [IDK] token in cases of uncertainty. We hypothesize that this generalizes to uncertainty expressed on downstream tasks like Q&A, which we experimentally verify later on.

We next describe in detail the technicalities of the [IDK] token and our training method.

2.1 The [IDK] token

The purpose of the new token is to represent lack of knowledge. Ideally, whenever the model would have been making a mistake, we want it to instead predict this token. That is, rather than generating a wrong token, we would like to model to generate the [IDK] token, as a means of conveying its uncertainty. We can consider this as a model expressing its lack of knowledge and may then choose to ignore its outputs. The more the model opts for this token rather than predicting the wrong answer, the more we improve the model's precision.

For instance, let us consider the setup of Factual Sentence Completion. In this setup, the model receives an incomplete sentence as an input and is expected to complete it factually. For example, a valid input would be "Paris is the capital of", and a factually correct output by the model would be "France". In this setup, if the model was going to predict "Germany", using the [IDK] token instead increases factual precision by refusing to answer a question where the answer would have been wrong. Naturally, almost universally predicting [IDK] indiscriminately may yield high precision but is not helpful. Therefore, taking into account the recall of factually correct answers is crucial in evaluating our method. We analyze both the precision and recall of our method in Section 4.

We add this new [IDK] token to the model's vocabulary and initialize its embedding randomly. The embedding is optimized alongside the rest of the model's parameters during training. We next describe our proposed IDK objective.

2.2 The IDK Training Objective

We modify the conventional cross-entropy objective between the softmax distribution over the model's prediction and the correct answer, such that each time the model fails to predict the correct token, it is encouraged to instead put some probability mass on the [IDK]. This encouragement is modulated by an *Uncertainty Factor* denoted as $\lambda \in [0, 1]$ that is larger the more uncertain the model is and exactly 0 when the model predicts the correct token.

We now define our modified cross-entropy objective. We use [gold] to denote the gold token (correct target) for each prediction. We denote the probability mass assigned to an arbitrary token [tok] in the prediction of a model as $\texttt{prob}(y_t = \texttt{[tok]}|y_{<t}, x)$ We further use $\mathbf{1}_{\texttt{[IDK]}}$ to denote a one-hot target vector with one at the index of the [IDK] token. Per convention, $\mathbf{y}$ denotes the one-hot target vector for the [gold] token. The modified objective is defined as follows:

$$\mathcal{L}_{\texttt{IDK}} = \mathcal{L}_{\mathrm{CE}}(\hat{\mathbf{y}},\, (1-\lambda)\,\mathbf{y} + \lambda\,\mathbf{1}_{\texttt{[IDK]}}) \tag{1}$$

If the model is uncertain in its prediction, the target is shifted away from predicting the [gold] token and towards the [IDK] token. This is modulated by λ. Note that in case the model makes the correct prediction, $\lambda = 0$ and $\mathcal{L}_{\texttt{IDK}}$ therefore reduces to the regular cross-entropy loss. When the model is correct, $\mathcal{L}_{\texttt{IDK}}$ simply provides the signal for the correct prediction. When the model is incorrect, $\mathcal{L}_{\texttt{IDK}}$ provides both the signal for the correct prediction and a signal to express uncertainty. We now detail the construction of the *Uncertainty Factor* λ.

The *Uncertainty Factor*. λ is constructed as a scalar weight with $\lambda \in [0, 1]$. Intuitively, we want λ to be close to 1 when the model is very uncertain and 0 when the model makes the correct prediction. Based on this, we define λ as one minus the probability mass on the gold token divided by the maximum probability mass put on any token:

$$\lambda = \Pi \times \left(1 - \frac{\texttt{prob}(y_t = \texttt{[gold]}|y_{<t}, x)}{\max_{\mathrm{i}}(\texttt{prob}(y_t = i|y_{<t}, x))}\right), \tag{2}$$

where $\Pi \in [0, 1]$ is a hyperparameter to control the influence of our objective. When the gold token probability is close to the maximum probability, λ is close to 0. If the model makes a correct prediction (the gold token is assigned the maximum probability), λ is 0, thereby reducing Equation 1 to the regular cross-entropy loss. When the gold token probability is much lower than the maximum probability, λ is close to 1, which translates to shifting almost all the probability mass of the target in Equation 1 to the [IDK] token. Π specifies the upper bound of target probability mass that can be shifted to the [IDK] token. For example, $\Pi = \frac{1}{2}$ means that at most half of the probability mass in the target can be shifted to [IDK] while the rest remains with the gold token. In practice, we do not tune this and set $\Pi = \frac{1}{2}$. This prevents the [IDK] token from ever becoming a better prediction than the gold token while still providing enough signal to predict [IDK] for uncertain predictions. We perform an ablation of the influence of Π in Section 4.2.

Uncertainty Regularization. An important consideration in designing the $\mathcal{L}_{\texttt{IDK}}$ objective is to prevent a collapse where the model is miscalibrated with too many false positive [IDK]s, putting too much probability mass on [IDK], although it could have made the correct prediction. Therefore, we add the following anti-false positive regularization to our objective:

$$\mathcal{L}_{\texttt{FP-reg}} = -\log(1 - \texttt{prob}(y_t = \texttt{[IDK]}|y_{<t}, x)), \tag{3}$$

which is exactly the binary cross-entropy objective with 0 as the target and the probability mass assigned to the [IDK] as the input. We only add this regularization objective when the model's prediction is correct. This aims to minimize the [IDK] token's probability mass the model learns to predict in cases it knows the answer – thus teaching it to minimize the use of this token in cases it is more certain, and is designed to reduce a decrease of its recall. We perform an ablation of $\mathcal{L}_{\texttt{FP-reg}}$ in Section 4.2.

The final loss. Combining all objectives, our final IDK objective is therefore:

$$\mathcal{L} = \begin{cases} \mathcal{L}_{\text{CE}} + \mathcal{L}_{\texttt{FP-reg}} & \text{if } \lambda = 0 \\ \mathcal{L}_{\texttt{IDK}} & \text{otherwise.} \end{cases} \tag{4}$$

3 Experiments

We use our proposed IDK objective to tune various pretrained models to use the new [IDK] token. We dub this process IDK-tuning. We then report the results of the IDK-tuned models on commonly used factual benchmarks, showing that our method improves factuality while paying only a small price in terms of knowledge recall. We also show that model size plays a significant role in the success of our method to create an effective uncertainty-aware model.

We employ *continual training* of pretrained models rather than training from scratch for two reasons: (i) the computational cost of training models that perform competitively on current benchmarks from scratch would be prohibitive, and (ii) starting from a model that is already a strong language modeler helps during the optimization process by providing a rough initial calibration that we utilize to derive the *Uncertainty Factor*.

3.1 IDK-tuning Setup

We use bert-base-cased [Devlin et al., 2019], mistralai/Mistral-7B-v0.1 [Jiang et al., 2023], and EleutherAI/pythia-70m – 2.8B [Biderman et al., 2023] for our base models for IDK-tuning. For IDK-tuning Mistral-7B-v0.1, we train on data randomly sampled from The Pile [Gao et al., 2020][2] with a context length of 4,096. We use example packing to fill the entire context length. We use a maximum learning rate of 4×10^{-5} with a linear warmup for 10% of the training steps and a cosine decay down to 2×10^{-6}. We use a batch size of 256, weight decay of 0.05, gradient clipping of 1.0 and AdamW betas (0.9, 0.95). We train for 1,024 optimizer steps resulting in a total of 1B training tokens. For the pythia-70m – 2.8B models, we use the same hyperparameters but reduce the context length to 2,048 to match the model's positional embeddings. We use bfloat16 and float16 mixed-precision training to match Mistral-7B-v0.1 and pythia-410m – 2.8B pretraining, respectively. For pythia-70m, pythia-160m and bert-base-cased, we observed NaN errors in the predicted logits irrespective of our loss modifications. Since the models are small enough, we switch to pure float32 for these models without using mixed-precision. In addition, for bert-base-cased we apply MLM [Devlin et al., 2019], while for each input, we randomly mask one of the tokens.

[2] We use monology/pile-uncopyrighted on the Huggingface Hub for a version of The Pile without the Books corpus, which contains copyrighted works.

3.2 Evaluation Setup

Evaluation Data. We consider the following datasets: LAMA [Petroni et al., 2019], TriviaQA [Joshi et al., 2017], and PopQA [Mallen et al., 2022]. These cover a wide range of queries, for example trivia questions (TriviaQA), and subject-relation-object facts phrased as queries (LAMA, PopQA). We consider the closed-book open-ended setting, where we do not provide any context or answer choices to the model. Importantly, in the case of TriviaQA and PopQA, where the input is formed as a question, we reduce it into a sentence completion task, using GPT4. Specifically, we prompt it to phrase the question as a sentence, while also providing it with some in-context examples that we manually created. See Appendix C for more details and the full prompt. To evaluate multiple-choice question answering, we use EleutherAI's `lm-evaluation-harness` [Gao et al., 2023]. Specifically, we use ARC [Clark et al., 2018], HellaSwag [Zellers et al., 2019], MMLU [Hendrycks et al., 2020], TruthfulQA [Lin et al., 2022a], WinoGrande [Sakaguchi et al., 2021], and GSM8k [Cobbe et al., 2021].

Baselines. For each of the evaluation datasets, we compare the IDK-tuned model with its original base model without any further training. Furthermore, we consider three different baselines:

1. **Confidence Threshold** baseline: We use the predicted probability mass in the language modeling head of the LM as a measure of confidence in the prediction [Yoshikawa and Okazaki, 2023]. We consider the first token generated by the LM. In case the corresponding probability mass of this token is greater than a fixed threshold, we consider the generation as valid. Otherwise, we consider this as an uncertainty expression (analogous to an [IDK] token generation in our model). To create a strong baseline, we search for the best threshold via hyperparameter tuning on the development set.
2. **P(True)** baseline [Kadavath et al., 2022]: Given an input sentence to complete, which we refer to as I, we use the original model to generate the completion, which we refer to as A. We then concatenate I and A and ask the model: *"Please answer either with 'true' or 'false' only. Is it true that: IA"*. If the model answer is not 'true', we consider this specific example as unknown for the model – namely the same case as if the IDK-tuned model would generate the [IDK].
3. **Semantic Entropy** baseline [Kuhn et al., 2023, Aichberger et al., 2024]: We sample K text generations from the model, encode them using a state-of-the-art semantic encoder and cluster their encodings. If the largest cluster size is larger than $\frac{K}{2}$, then we take a random generation out of this cluster as the model's answer. Otherwise, we consider this example as unknown.

Evaluation. We evaluate how well our models use the new [IDK] token by measuring their factuality and knowledge memory, using the following metrics: (i) **Precision**: the portion of factually correct completions, out of all the claims that have been completed with any token that is different from the [IDK] token, i.e., the claims that the model was certain enough about, and tried to factually complete. (ii) **Recall**: the portion of factually correct completions, out of all the claims in the dataset. Namely, the portion of knowledge memory the model has, out of the entire test set we evaluate on. (iii) **F1**: the harmonic mean of precision and recall. In the case of base models without additional calibration methods, the precision, recall, and F1-scores all correspond to their accuracy on the task.

In Section 4.2, we use two further metrics to analyze the patterns when IDK-tuned models predict [IDK]. For this, we use the notion of correctly predicting [IDK]: We consider an [IDK] prediction to be correct if the base model does not predict the correct answer for an instance. We define (i) **IDK recall**: the fraction of instances the model predicted [IDK] correctly out of all instances where the base model did in fact not predict the correct answer, and (ii) **IDK error rate**: the fraction of instances where the model predicted [IDK] incorrectly out of all instances where the base model did indeed predict the correct answer.

4 Results

We next report results showing that our proposed IDK-tuning method can effectively improve factuality while causing only a small loss of existing knowledge.

	LAMA Google-RE			LAMA T-Rex			LAMA SQuAD			TriviaQA			PopQA		
	P	**R**	**F1**	**P**	**R**	**F1**	**P**	**R**	**F1**	**P**	**R**	**F1**	**P**	**R**	**F1**
`Mistral-7B-v0.1`	48.1	48.1	48.1	71.2	**71.2**	71.2	45.8	45.8	45.8	52.0	52.0	52.0	35.5	**35.5**	**35.5**
`Mistral-7B-v0.1` + The Pile	48.8	**48.8**	48.8	69.9	69.9	69.9	48.0	**48.0**	48.0	52.2	**52.2**	52.2	35.2	35.2	35.2
`Mistral-7B-v0.1` + Confidence Threshold	60.0	40.0	48.0	80.4	63.5	71.0	64.4	33.5	44.1	70.4	41.1	51.9	64.6	20.6	31.2
`Mistral-7B-v0.1` + P(True)	54.4	44.5	48.9	73.8	65.1	69.2	54.9	41.0	46.9	58.8	47.5	52.5	40.3	29.0	33.7
`Mistral-7B-v0.1` + Semantic Entropy	70.1	38.9	50.0	88.0	65.4	75.0	70.2	44.5	54.4	68.5	52.5	59.4	68.7	20.4	31.5
`Mistral-7B-v0.1` + `IDK`-tuning on The Pile	**71.1**	40.6	**51.7**	**88.5**	65.5	**75.3**	**72.0**	44.3	**54.9**	**72.5**	52.0	**60.6**	**78.1**	20.5	32.5

Table 1: Precision (P), Recall (R), and F1-scores for `Mistral-7B-v0.1`. Our `IDK`-tuning achieves the best precision with minor decreases in recall, outperforming previous work. `Mistral-7B-v0.1` + Confidence Threshold refers to the baseline based on the probability mass of the predicted answer [Yoshikawa and Okazaki, 2023]. `Mistral-7B-v0.1` + The Pile refers to the ablation discussed in Section 4.2.

	P	**R**	**F1**
`Mistral-7B-v0.1`	28.2	**28.2**	28.2
`Mistral-7B-v0.1` + The Pile	28.3	28.3	28.3
`Mistral-7B-v0.1` + Confidence Threshold	45.0	18.5	26.2
`Mistral-7B-v0.1` + `IDK`-tuning on The Pile	**48.8**	20.8	**29.2**

Table 2: Precision (P), Recall (R), and F1-scores of our model on the `lm-eval-harness`, compared to baselines.

4.1 Main Results

`Mistral-7B-v0.1` results. Table 1 shows the results of our largest model `Mistral-7B-v0.1` on factual closed-book sentence completion datasets. Our results show that the `IDK`-tuned `Mistral-7B-v0.1` has a much higher precision – namely the model generates significantly fewer factually incorrect completions and instead puts probability mass on the `[IDK]` token. However, the model does show decreased knowledge recall on some tasks. Overall, we observe an increase in the average F1-score. Table 2 shows the averaged results on the `lm-eval-harness` datasets. The trend here is similar, although the increase in precision compared to baselines is slightly lower. This suggests that the model tends to be more certain when it comes to multiple-choice questions.

Scaling behavior of `IDK`-tuning. We further investigate the effect of model size on the success of `IDK`-tuning. We conduct `IDK`-tuning for each of the `pythia-70m` – `2.8B` models as detailed in Section 3.1. In Figure 2, we plot the average precision, recall, and F1-score for each of `pythia-70m` – `2.8B` as well as `Mistral-7B-v0.1`, over all the closed-book sentence completion datasets. We observe a clear trend of recall and F1-score increasing log-linearly with the model size. The precision of `IDK`-tuned models increases only slightly as the model size increases. For the two smallest models we investigate (`pythia-70m` and `pythia-160m`), our method is arguably not effective, as the `IDK`-tuned model's recall collapses (we further analyze this in Section 4.3).

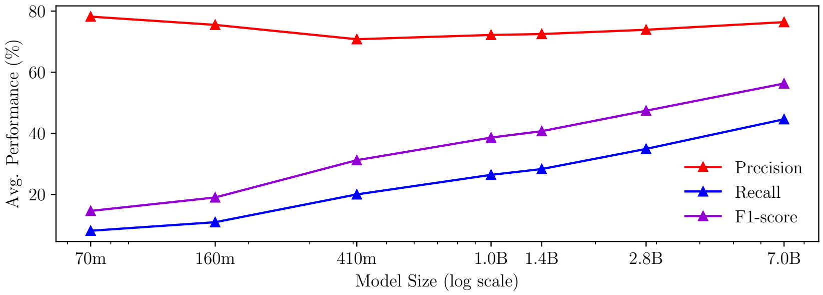

Figure 2: Average performance on closed-book factual sentence completion benchmarks of `IDK`-tuned models in terms of their parameter count. 70m to 2.8B are `pythia-70m` – `2.8B`, while 7.0B is `Mistral-7B-v0.1`.

	LAMA Google-RE			LAMA T-Rex			LAMA SQuAD		
	P	**R**	**F1**	**P**	**R**	**F1**	**P**	**R**	**F1**
`bert-base-cased`	23.0	**23.0**	23.0	59.8	**59.8**	59.8	9.5	**9.5**	9.5
`bert-base-cased` + Confidence Treshold	58.8	15.8	24.9	71.5	35.9	47.8	69.5	5.0	9.3
`bert-base-cased` + `IDK`-tuning	**78.1**	15.9	**26.4**	**72.5**	53.0	**61.2**	**80.2**	6.4	**11.9**

Table 3: Precision (P), Recall (R), and F1 scores for of our `IDK`-tuned `bert-base-cased` on the evaluation benchmarks, compared to baselines.

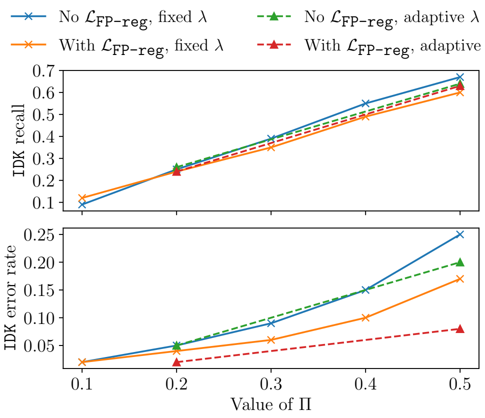

Figure 3: Ablation study of different values for the Π factor that controls the upper bound of probability mass put on `[IDK]` in the target.

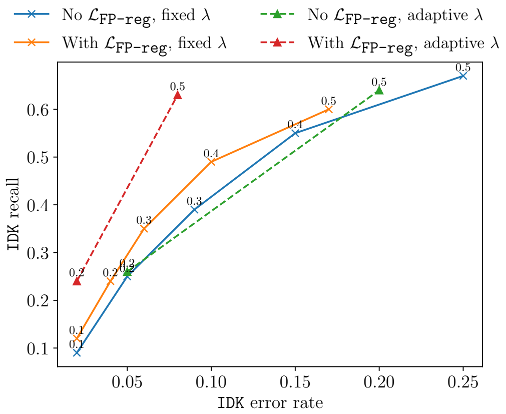

Figure 4: Tradeoff between `IDK` recall and `IDK` error rate for different parameter combinations. We annotate each data point with its corresponding Π value.

`bert-base-cased` results. Table 3 reports the results of out `IDK`-tuned `bert-base-cased` model. We see a similar trend as in our evaluation of `Mistral-7B-v0.1`. Factuality is improved, while recall is reduced by only a small amount.

4.2 Ablations

We perform an ablation study of our method to further investigate the effectiveness of each of its components. For our study, we calculate the `IDK` recall and `IDK` error rate on the closed-book factual sentence completion datasets. We study the effect of Π, λ and the $\mathcal{L}_{\texttt{FP-reg}}$ term. For this, we perform `IDK`-tuning using `Mistral-7B-v0.1` with the same hyperparameters as our main runs with different combinations of the studied components[3]. We plot the `IDK` recall for different values of Π in Figure 3. In Figure 4, we plot the `IDK` recall vs. `IDK` error rate tradeoff. `IDK` recall and `IDK` error rate are defined in Section 3.2. We study different aspects of these results below:

1. Analysis of the adaptive nature of the *Uncertainty Factor* λ. The *Uncertainty Factor* λ defined in Equation 2 is *adaptive*, meaning the amount of probability mass shifted to `[IDK]` depends on the predicted probability distribution. Another possible choice is to use a *fixed* $\lambda \in [0, 1]$. We analyze this in Figure 3 and Figure 4[4]. We can see that using the adaptive λ formulation results in a lower `IDK` error rate without a major decrease in `IDK` recall.

2. Effect of the $\mathcal{L}_{\texttt{FP-reg}}$ regularization. We also study the effect of the $\mathcal{L}_{\texttt{FP-reg}}$ term (see Section 2.2). Again, we see that using $\mathcal{L}_{\texttt{FP-reg}}$ results in a reduced `IDK` error rate without decreasing `IDK` recall significantly.

[3] Due to computational constraints, we run this for a reduced set of Π for the cases with adaptive λ.

[4] For a fixed λ, we set $\lambda = \Pi$.

3. Effect of the upper bound hyperparameter Π. We also study the effect of Π, which is the upper bound of the *Uncertainty Factor* (see Equation 2). Our ablation study demonstrates that increasing Π results in an increase in correct predictions of [IDK] (higher IDK recall), at the cost of a small increase of erroneous [IDK] predictions (IDK error rate). The IDK error rate increases less when using both our proposed adaptive λ and $\mathcal{L}_{\text{FP-reg}}$.

Effect of knowledge contained in The Pile. Since we conduct further pretraining on The Pile, improved performance of our method could be partly explained by additional knowledge that the model learns during IDK-tuning. However, we show that this is not the case. In the case of the pythia-70m – 2.8B models, our data used for IDK-tuning exactly matches their pretraining data. For Mistral-7B-v0.1, this is not known although The Pile was likely also included. We note that the language modeling performance on The Pile of our models during IDK-tuning actually very slightly decreases rather than improving, suggesting the absence of any newly learned knowledge. However, to completely rule out any such effects, we trained Mistral-7B-v0.1 on the exact sample of The Pile used for IDK-tuning but with the regular cross-entropy objective. We report the performance of this model in Table 1. Indeed, Mistral-7B-v0.1 with further training on The Pile performs similarly to the base Mistral-7B-v0.1 on average.

4.3 Analysis of Optimization Stability

Collapse to [IDK]. Highly optimizing every component of the standard language modeling task with Transformers has made it easy to forget that optimization processes of deep neural networks can be brittle and divergent. Naively replacing the regular cross-entropy objective with our IDK-loss $\mathcal{L}_{\text{IDK}}$ leads to a collapse of training where the model simply always learns to put most probability mass on the [IDK]. We already account for this by (i) introducing the Π inhibitor to allow us to set an upper bound on the maximum probability mass that is assigned to the [IDK] in the target vector and (ii) introducing the additional $\mathcal{L}_{\text{FP-reg}}$ regularization to provide an additional signal that punishes probability mass being assigned to [IDK] when the model's prediction is already correct.

In practice, we see that the regular cross-entropy loss shows a small uptick at the very beginning of IDK-tuning. In almost all runs, this recovers quickly back to baseline levels, where it remains. We find that with $\Pi = 0.5$ and the $\mathcal{L}_{\text{FP-reg}}$ regularization, most training runs are stable without further model-specific tuning.

Collapse for small models pythia-70m and pythia-160m. However, for pythia-160m and pythia-70m, which are the only runs in our experiments that diverge even with our added regularization losses, the regular cross-entropy keeps on rising with a large spike. Concretely, the predicted distributions not only show an increased cross-entropy with the targets but also a sharply increasing entropy: we observe that the predicted distributions collapse towards a uniform distribution. At the worst point, 0% of the predictions of pythia-160m are correct. However, both models somewhat recover towards the end of training but stay well below baseline levels in terms of language modeling performance. We note that this is a different collapse pattern than the collapse towards almost always predicting [IDK] observed without our regularization terms.

We further analyzed this and observe that for both pythia-160m and pythia-70m, the initial probability mass assigned to the [IDK] token is so small that it gets rounded to zero even when using float32 precision. This causes the $\mathcal{L}_{\text{IDK}}$ loss to be very large, resulting in large gradient norms. Already for pythia-410m, the initial probability mass on [IDK] is substantial enough to prevent this (albeit still a very small value smaller than $5 \times 10^{-}9$). Both pythia-160m and pythia-70m also show a larger initial entropy in their predicted distributions (i.e., "flatter" predicted distributions). We conjecture that an adapted initialization of the [IDK] token and/or a small bias towards [IDK] at the beginning of training could prevent this divergence. As we only encounter this issue for the small pythia-160m and pythia-70m models, we leave further investigation of this for future work.

4.4 Text Generation

To assess whether our IDK-tuning might harm other different downstream language skills, which are not necessarily only factual, we evaluate the IDK-tuned Mistral-7B-v0.1 on the task of text summarization, and compare its results to those of the original model. For this, due to the high likelihood of the [IDK] token being generated during a longer text generation process, we use greedy decoding and ignore the [IDK] token. For this experiment, we use four different common

	Legal Plain English	TLDR	SPEC5G
Mistral-7B-v0.1	**17.5**	**14.1**	37.2
Mistral-7B-v0.1 + The Pile	17.4	**14.1**	**37.3**
Mistral-7B-v0.1 + IDK-tuning on The Pile	17.2	14.0	36.9

Table 4: RougeL scores on different summarization tasks to measure the impact of IDK-tuning on other language model abilities. Mistral-7B-v0.1 + The Pile refers to the ablation discussed in Section 4.2.

	No effect	Noise	White Noise	Abstaining
Mistral-7B-v0.1	68.5%	9%	6.5%	16%
pythia-2.8B	59.5%	13.5%	12.5%	14.5%
pythia-70m	52%	18.5%	22%	7.5%

Table 5: Error type distribution on 200 failures of our IDK-tuned models.

summarization benchmarks: Legal Plain English [Manor and Li, 2019], TLDR [Völske et al., 2017], and SPEC5G [Karim et al., 2023]. We measure performance using RougeL [Lin, 2004], as it is widely used in related work, and report the results in Table 4. The IDK-tuned Mistral-7B-v0.1 performs only slightly worse than the original base model. This is an encouraging result, as it means that IDK-tuning does not necessarily harm other language skills of pretrained language models.

4.5 Error Analysis

To gauge the effect of IDK-tuning on model responses to factual prompts and questions, we conduct an in-depth manual analysis on a random sample of 200 (40 from each dataset) of the model's incorrect generations (generations that do not contain the correct answer). We conduct this analysis for three models across model sizes: pythia-70m, pythia-2.8B, and Mistral-7B-v0.1. We then categorize each of these incorrect generations to one of the following categories:

1. *No Effect*: Both the original model and the IDK-tuned model generate the same (incorrect) answer.
2. *Noise*: The original model generates the correct answer, while the IDK-tuned model does not.
3. *White Noise*: Both the original and IDK-tuned models do not generate the correct answer, however the IDK-tuned model generates a different one.
4. *Abstain*: The IDK-tuned model abstains from answering by generating text such as “unknown" or “mystery".

The results are shown in Table 5. Our analysis suggest that first, the bigger the model, the fewer changes our training approach causes in the model's generations, and second, the bigger the model, the greater its ability to abstain from answering via words (which generally can be interpreted as equal to generating an [IDK] token, although harder to evaluate automatically).

5 Related Work

Model Calibration. Our goal is closely related to the key challenge of model calibration [Guo et al., 2017b]: to provide a measure of the probability that a prediction is incorrect alongside the actual prediction. The problem of factual error detection can be viewed as a variation of calibration, where instead of a continuous probability, we provide a binary prediction for whether the model is correct or not. This is also related to the setting of selective prediction, where models can abstain from answering a query [Varshney et al., 2022, Kamath et al., 2020]. Common approaches to calibration are to perform various transformations on a model's output logits [Desai and Durrett, 2020, Jiang et al., 2021], and measuring uncertainty [e.g., see Kuhn et al., 2023]. More recent works have studied the use of LMs for providing calibration, by training them on statements known to be factually correct or incorrect. This "supervised" approach has been explored via fine-tuning [Kadavath et al., 2022, Lin et al., 2022b], in-context learning [Cohen et al., 2023a, Alivanistos et al., 2022], zero-shot instruction-oriented [Cohen et al., 2023b] and consistency sampling [Yoran et al., 2023] techniques. Further recent studies [Azaria and Mitchell, 2023] use the internal state of the model for classifying whether it is certain or not, use a new token for unanswerable inputs [Lu et al., 2022], or construct a specific dataset for effectively tuning the model for answering refusal [Zhang et al., 2024]. Our work builds upon this, aiming to teach the model to assess and express its own uncertainty via the new `[IDK]` token we introduced.

Attribution. Another related line of work focuses on checking whether LM-generated texts are faithful to a given source text [Bohnet et al., 2022, Honovich et al., 2022]. This problem has been addressed via several approaches, including question generation [Wang et al., 2020, Honovich et al., 2021, Scialom et al., 2021], NLI [Thorne et al., 2018, Welleck et al., 2019, Maynez et al., 2020, Dziri et al., 2022, Gao et al., 2022, Kamoi et al., 2023], data augmentation [Atanasova et al., 2022, Wright et al., 2022, Gekhman et al., 2023], and planning schemes that allow the model to self-edit its own generation [Schick et al., 2022]. Unlike these works, we are not assuming any reference text or external knowledge bases. Instead, we aim to teach the model to decide on its own whether it is likely to be able to factually complete a sentence correctly.

6 Conclusion

We propose a novel method for improving LMs' factuality by adding a special `[IDK]` token to an LM's vocabulary. Alongside the new `[IDK]` token, we introduce a novel pretraining objective called `IDK`-tuning to model uncertainty in the model's prediction as the probability mass assigned to the `[IDK]`. Crucially, `IDK`-tuning requires no labeled data and is instead a drop-in replacement of the conventional cross-entropy loss used for self-supervised language modeling on web-crawled texts. This allows us to explore uncertainty-aware training at a large scale. In our experiments, we conduct continued pretraining of a diverse range of pretrained models using the `IDK` objective.

Evaluation on factual sentence completion and multiple-choice benchmarks shows that `IDK`-tuned models can complete these tasks with much higher precision by refusing to answer (assigning high probability mass to the `[IDK]` token) in cases when the base model would have given a wrong answer. This comes at only small decreases in recall. We investigate the scaling behavior of our method with respect to model size using the Pythia model suite [Biderman et al., 2023], perform several ablation studies for individual components of our `IDK` objective, and verify that the general language modeling ability of `IDK`-tuned models does not degrade.

Our work can be extended in several ways. For example, since we do not rely on any labels of our training data used for `IDK`-tuning, we potentially apply our objective for next-token predictions where it might be ill-posed. Instead, we can perform lightweight filtering of relevant next-token predictions, such as named entities, focusing our objective more on factual next-token predictions. Also, `IDK`-tuning can be applied during pretraining from scratch, where our `IDK` objective could have interesting interactions with the acquisition of new knowledge during this stage.

7 Limitations

We note a few limitations of our proposed method. First, it requires a full pretraining of LMs on relatively large corpus. This of course is both highly computationally expensive and time-consuming. It is likely often the case that this kind of training cannot be conducted on typical academic lab resources, on a large enough model, in a reasonable amount of time.

Second, as discussed in Section 4.4, our method may slightly harm certain language skills, such as long text generation. Other downstream skills may be affected more significantly. We further discuss potential risk and biases in Appendix A.

Acknowledgements

Roi Cohen and Gerard de Melo received funding from The Goldman Sachs Group, Inc., New York, NY, USA. Konstantin Dobler thanks the German Federal Ministry for Education and Research

(BMBF) through the project «KI-Servicezentrum Berlin Brandenburg» (01IS22092) and the European Laboratory for Learning and Intelligent Systems (ELLIS) PhD program for support. We further express our gratitude to the NeurIPS 2024 reviewers for their helpful comments.

References

Lukas Aichberger, Kajetan Schweighofer, Mykyta Ielanskyi, and Sepp Hochreiter. Semantically diverse language generation for uncertainty estimation in language models. *arXiv preprint arXiv:2406.04306*, 2024.

Dimitrios Alivanistos, Selene Báez Santamaría, Michael Cochez, Jan-Christoph Kalo, Emile van Krieken, and Thiviyan Thanapalasingam. Prompting as probing: Using language models for knowledge base construction. *arXiv preprint arXiv:2208.11057*, 2022.

Pepa Atanasova, Jakob Grue Simonsen, Christina Lioma, and Isabelle Augenstein. Fact checking with insufficient evidence. *Transactions of the Association for Computational Linguistics*, 10:746–763, 2022. doi: 10.1162/tacl_a_00486. URL `https://aclanthology.org/2022.tacl-1.43`.

Amos Azaria and Tom Mitchell. The internal state of an LLM knows when it's lying. In Houda Bouamor, Juan Pino, and Kalika Bali, editors, *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 967–976, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.findings-emnlp.68. URL `https://aclanthology.org/2023.findings-emnlp.68`.

Stella Biderman, Hailey Schoelkopf, Quentin Anthony, Herbie Bradley, Kyle O'Brien, Eric Hallahan, Mohammad Aflah Khan, Shivanshu Purohit, USVSN Sai Prashanth, Edward Raff, Aviya Skowron, Lintang Sutawika, and Oskar Van Der Wal. Pythia: a suite for analyzing large language models across training and scaling. In *Proceedings of the 40th International Conference on Machine Learning*, ICML'23. JMLR.org, 2023.

Bernd Bohnet, Vinh Q Tran, Pat Verga, Roee Aharoni, Daniel Andor, Livio Baldini Soares, Jacob Eisenstein, Kuzman Ganchev, Jonathan Herzig, Kai Hui, et al. Attributed question answering: Evaluation and modeling for attributed large language models. *arXiv preprint arXiv:2212.08037*, 2022.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901, 2020.

Miles Brundage, Shahar Avin, Jasmine Wang, Haydn Belfield, Gretchen Krueger, Gillian K. Hadfield, Heidy Khlaaf, Jingying Yang, Helen Toner, Ruth Fong, Tegan Maharaj, Pang Wei Koh, Sara Hooker, Jade Leung, Andrew Trask, Emma Bluemke, Jonathan Lebensbold, Cullen O'Keefe, Mark Koren, Théo Ryffel, JB Rubinovitz, Tamay Besiroglu, Federica Carugati, Jack Clark, Peter Eckersley, Sarah de Haas, Maritza L. Johnson, Ben Laurie, Alex Ingerman, Igor Krawczuk, Amanda Askell, Rosario Cammarota, Andrew J. Lohn, David Krueger, Charlotte Stix, Peter Henderson, Logan Graham, Carina E. A. Prunkl, Bianca Martin, Elizabeth Seger, Noa Zilberman, Se'an 'O h'Eigeartaigh, Frens Kroeger, Girish Sastry, Rebecca Kagan, Adrian Weller, Brian Tse, Elizabeth Barnes, Allan Dafoe, Paul Scharre, Ariel Herbert-Voss, Martijn Rasser, Shagun Sodhani, Carrick Flynn, Thomas Krendl Gilbert, Lisa Dyer, Saif Khan, Yoshua Bengio, and Markus Anderljung. Toward trustworthy AI development: Mechanisms for supporting verifiable claims. *ArXiv*, abs/2004.07213, 2020. URL `https://api.semanticscholar.org/CorpusID:215768885`.

Yangyi Chen, Lifan Yuan, Ganqu Cui, Zhiyuan Liu, and Heng Ji. A close look into the calibration of pre-trained language models. In Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki, editors, *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1343–1367, Toronto, Canada, July 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.acl-long.75. URL `https://aclanthology.org/2023.acl-long.75`.

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. Think you have solved question answering? try ARC, the AI2 reasoning challenge. *arXiv preprint arXiv:1803.05457*, 2018.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. Training verifiers to solve math word problems. *ArXiv*, abs/2110.14168, 2021. URL https://api.semanticscholar.org/CorpusID:239998651.

Roi Cohen, Mor Geva, Jonathan Berant, and Amir Globerson. Crawling the internal knowledge-base of language models. In Andreas Vlachos and Isabelle Augenstein, editors, *Findings of the Association for Computational Linguistics: EACL 2023*, pages 1856–1869, Dubrovnik, Croatia, May 2023a. Association for Computational Linguistics. doi: 10.18653/v1/2023.findings-eacl.139. URL https://aclanthology.org/2023.findings-eacl.139.

Roi Cohen, May Hamri, Mor Geva, and Amir Globerson. LM vs LM: Detecting factual errors via cross examination. In Houda Bouamor, Juan Pino, and Kalika Bali, editors, *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 12621–12640, Singapore, December 2023b. Association for Computational Linguistics. doi: 10.18653/v1/2023.emnlp-main.778. URL https://aclanthology.org/2023.emnlp-main.778.

Roi Cohen, Eden Biran, Ori Yoran, Amir Globerson, and Mor Geva. Evaluating the ripple effects of knowledge editing in language models. *Transactions of the Association for Computational Linguistics*, 12:283–298, 2024.

Shrey Desai and Greg Durrett. Calibration of pre-trained transformers. In Bonnie Webber, Trevor Cohn, Yulan He, and Yang Liu, editors, *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 295–302, Online, November 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-main.21. URL https://aclanthology.org/2020.emnlp-main.21.

Ashwin Devaraj, William Sheffield, Byron Wallace, and Junyi Jessy Li. Evaluating factuality in text simplification. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 7331–7345, Dublin, Ireland, May 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.acl-long.506. URL https://aclanthology.org/2022.acl-long.506.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: Pre-training of deep bidirectional transformers for language understanding. In *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)*, pages 4171–4186, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/N19-1423. URL https://aclanthology.org/N19-1423.

Nouha Dziri, Hannah Rashkin, Tal Linzen, and David Reitter. Evaluating attribution in dialogue systems: The BEGIN benchmark. *Transactions of the Association for Computational Linguistics*, 10:1066–1083, 2022. doi: 10.1162/tacl_a_00506. URL https://aclanthology.org/2022.tacl-1.62.

Yanai Elazar, Nora Kassner, Shauli Ravfogel, Abhilasha Ravichander, Eduard Hovy, Hinrich Schütze, and Yoav Goldberg. Measuring and improving consistency in pretrained language models. *Transactions of the Association for Computational Linguistics*, 9:1012–1031, 2021. doi: 10.1162/tacl_a_00410. URL https://aclanthology.org/2021.tacl-1.60.

Leo Gao, Stella Biderman, Sid Black, Laurence Golding, Travis Hoppe, Charles Foster, Jason Phang, Horace He, Anish Thite, Noa Nabeshima, Shawn Presser, and Connor Leahy. The Pile: An 800GB dataset of diverse text for language modeling, 2020.

Leo Gao, Jonathan Tow, Baber Abbasi, Stella Biderman, Sid Black, Anthony DiPofi, Charles Foster, Laurence Golding, Jeffrey Hsu, Alain Le Noac'h, Haonan Li, Kyle McDonell, Niklas Muennighoff, Chris Ociepa, Jason Phang, Laria Reynolds, Hailey Schoelkopf, Aviya Skowron, Lintang Sutawika, Eric Tang, Anish Thite, Ben Wang, Kevin Wang, and Andy Zou. A framework for few-shot language model evaluation, 12 2023. URL https://zenodo.org/records/10256836.

Luyu Gao, Zhuyun Dai, Panupong Pasupat, Anthony Chen, Arun Tejasvi Chaganty, Yicheng Fan, Vincent Y Zhao, Ni Lao, Hongrae Lee, Da-Cheng Juan, et al. Rarr: Researching and revising what language models say, using language models. *arXiv preprint arXiv:2210.08726*, 2022.

Zorik Gekhman, Jonathan Herzig, Roee Aharoni, Chen Elkind, and Idan Szpektor. Trueteacher: Learning factual consistency evaluation with large language models. *arXiv preprint arXiv:2305.11171*, 2023.

Jiahui Geng, Fengyu Cai, Yuxia Wang, Heinz Koeppl, Preslav Nakov, and Iryna Gurevych. A survey of confidence estimation and calibration in large language models. In Kevin Duh, Helena Gomez, and Steven Bethard, editors, *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 6577–6595, Mexico City, Mexico, June 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.naacl-long.366. URL `https://aclanthology.org/2024.naacl-long.366`.

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q. Weinberger. On calibration of modern neural networks. In *Proceedings of the 34th International Conference on Machine Learning - Volume 70*, ICML'17, page 1321–1330. JMLR.org, 2017a.

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q. Weinberger. On calibration of modern neural networks. In Doina Precup and Yee Whye Teh, editors, *Proceedings of the 34th International Conference on Machine Learning*, volume 70 of *Proceedings of Machine Learning Research*, pages 1321–1330. PMLR, 06–11 Aug 2017b. URL `https://proceedings.mlr.press/v70/guo17a.html`.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding. *arXiv preprint arXiv:2009.03300*, 2020.

Or Honovich, Leshem Choshen, Roee Aharoni, Ella Neeman, Idan Szpektor, and Omri Abend. q^2: Evaluating factual consistency in knowledge-grounded dialogues via question generation and question answering. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7856–7870, Online and Punta Cana, Dominican Republic, November 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.emnlp-main.619. URL `https://aclanthology.org/2021.emnlp-main.619`.

Or Honovich, Roee Aharoni, Jonathan Herzig, Hagai Taitelbaum, Doron Kukliansy, Vered Cohen, Thomas Scialom, Idan Szpektor, Avinatan Hassidim, and Yossi Matias. True: Re-evaluating factual consistency evaluation. *arXiv preprint arXiv:2204.04991*, 2022.

Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Ye Jin Bang, Andrea Madotto, and Pascale Fung. Survey of hallucination in natural language generation. *ACM Comput. Surv.*, 55(12), mar 2023. ISSN 0360-0300. doi: 10.1145/3571730. URL `https://doi.org/10.1145/3571730`.

Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. Mistral 7b. *arXiv preprint arXiv:2310.06825*, 2023.

Zhengbao Jiang, Jun Araki, Haibo Ding, and Graham Neubig. How can we know when language models know? on the calibration of language models for question answering. *Transactions of the Association for Computational Linguistics*, 9:962–977, 2021. doi: 10.1162/tacl_a_00407. URL `https://aclanthology.org/2021.tacl-1.57`.

Mandar Joshi, Eunsol Choi, Daniel S Weld, and Luke Zettlemoyer. TriviaQA: A large scale distantly supervised challenge dataset for reading comprehension. *arXiv preprint arXiv:1705.03551*, 2017.

Saurav Kadavath, Tom Conerly, Amanda Askell, Tom Henighan, Dawn Drain, Ethan Perez, Nicholas Schiefer, Zachary Dodds, Nova Dassarma, Eli Tran-Johnson, Scott Johnston, Sheer El-Showk, Andy Jones, Nelson Elhage, Tristan Hume, Anna Chen, Yuntao Bai, Sam Bowman, Stanislav Fort, Deep Ganguli, Danny Hernandez, Josh Jacobson, John Kernion, Shauna Kravec, Liane Lovitt, Kamal Ndousse, Catherine Olsson, Sam Ringer, Dario Amodei, Tom B. Brown, Jack Clark, Nicholas Joseph, Benjamin Mann, Sam McCandlish, Christopher Olah, and Jared Kaplan. Language models (mostly) know what they know. *arXiv preprint arXiv:2207.05221*, 2022.

Amita Kamath, Robin Jia, and Percy Liang. Selective question answering under domain shift. *arXiv preprint arXiv:2006.09462*, 2020.

Ryo Kamoi, Tanya Goyal, and Greg Durrett. Shortcomings of question answering based factuality frameworks for error localization. In Andreas Vlachos and Isabelle Augenstein, editors, *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*, pages 132–146, Dubrovnik, Croatia, May 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.eacl-main.11. URL `https://aclanthology.org/2023.eacl-main.11`.

Imtiaz Karim, Kazi Samin Mubasshir, Mirza Masfiqur Rahman, and Elisa Bertino. SPEC5G: A dataset for 5G cellular network protocol analysis. In Jong C. Park, Yuki Arase, Baotian Hu, Wei Lu, Derry Wijaya, Ayu Purwarianti, and Adila Alfa Krisnadhi, editors, *Findings of the Association for Computational Linguistics: IJCNLP-AACL 2023 (Findings)*, pages 20–38, Nusa Dua, Bali, November 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.findings-ijcnlp.3. URL `https://aclanthology.org/2023.findings-ijcnlp.3`.

Lorenz Kuhn, Yarin Gal, and Sebastian Farquhar. Semantic uncertainty: Linguistic invariances for uncertainty estimation in natural language generation. *arXiv preprint arXiv:2302.09664*, 2023.

Chin-Yew Lin. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain, July 2004. Association for Computational Linguistics. URL `https://aclanthology.org/W04-1013`.

Stephanie Lin, Jacob Hilton, and Owain Evans. TruthfulQA: Measuring how models mimic human falsehoods. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3214–3252, Dublin, Ireland, May 2022a. Association for Computational Linguistics. doi: 10.18653/v1/2022.acl-long.229. URL `https://aclanthology.org/2022.acl-long.229`.

Stephanie Lin, Jacob Hilton, and Owain Evans. Teaching models to express their uncertainty in words. *arXiv preprint arXiv:2205.14334*, 2022b.

Hongyuan Lu, Wai Lam, Hong Cheng, and Helen Meng. On controlling fallback responses for grounded dialogue generation. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 2591–2601, Dublin, Ireland, May 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.findings-acl.204. URL `https://aclanthology.org/2022.findings-acl.204`.

Alex Mallen, Akari Asai, Victor Zhong, Rajarshi Das, Hannaneh Hajishirzi, and Daniel Khashabi. When not to trust language models: Investigating effectiveness and limitations of parametric and non-parametric memories. *arXiv preprint arXiv:2212.10511*, 2022.

Laura Manduchi, Kushagra Pandey, Robert Bamler, Ryan Cotterell, Sina Däubener, Sophie Fellenz, Asja Fischer, Thomas Gärtner, Matthias Kirchler, Marius Kloft, Yingzhen Li, Christoph Lippert, Gerard de Melo, Eric Nalisnick, Björn Ommer, Rajesh Ranganath, Maja Rudolph, Karen Ullrich, Guy Van den Broeck, Julia E Vogt, Yixin Wang, Florian Wenzel, Frank Wood, Stephan Mandt, and Vincent Fortuin. On the challenges and opportunities in Generative AI. *arXiv preprint arXiv:2403.00025*, 2024. URL `https://arxiv.org/abs/2403.00025`.

Laura Manor and Junyi Jessy Li. Plain English summarization of contracts. In *Proceedings of the Natural Legal Language Processing Workshop 2019*, pages 1–11, Minneapolis, Minnesota, June 2019. Association for Computational Linguistics. doi: 10.18653/v1/W19-2201. URL `https://aclanthology.org/W19-2201`.

Joshua Maynez, Shashi Narayan, Bernd Bohnet, and Ryan McDonald. On faithfulness and factuality in abstractive summarization. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1906–1919, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.173. URL `https://aclanthology.org/2020.acl-main.173`.

Niels Mündler, Jingxuan He, Slobodan Jenko, and Martin Vechev. Self-contradictory hallucinations of large language models: Evaluation, detection and mitigation. *arXiv preprint arXiv:2305.15852*, 2023.

Jeff Z. Pan, Simon Razniewski, Jan-Christoph Kalo, Sneha Singhania, Jiaoyan Chen, Stefan Dietze, Hajira Jabeen, Janna Omeliyanenko, Wen Zhang, Matteo Lissandrini, Russa Biswas, Gerard de Melo, Angela Bonifati, Edlira Vakaj, Mauro Dragoni, and Damien Graux. Large Language Models and Knowledge Graphs: Opportunities and Challenges. *Transactions on Graph Data and Knowledge*, 1(1):2:1–2:38, 2023. doi: 10.4230/TGDK.1.1.2. URL `https://drops.dagstuhl.de/entities/document/10.4230/TGDK.1.1.2`.

Fabio Petroni, Tim Rocktäschel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, Alexander H Miller, and Sebastian Riedel. Language models as knowledge bases? *arXiv preprint arXiv:1909.01066*, 2019.

Hannah Rashkin, Vitaly Nikolaev, Matthew Lamm, Lora Aroyo, Michael Collins, Dipanjan Das, Slav Petrov, Gaurav Singh Tomar, Iulia Turc, and David Reitter. Measuring attribution in natural language generation models. *Computational Linguistics*, 49(4):777–840, 2023.

Adam Roberts, Colin Raffel, and Noam Shazeer. How much knowledge can you pack into the parameters of a language model? *arXiv preprint arXiv:2002.08910*, 2020.

Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. WinoGrande: An adversarial winograd schema challenge at scale. *Commun. ACM*, 64(9):99–106, aug 2021. ISSN 0001-0782. doi: 10.1145/3474381. URL `https://doi.org/10.1145/3474381`.

Timo Schick, Jane Dwivedi-Yu, Zhengbao Jiang, Fabio Petroni, Patrick Lewis, Gautier Izacard, Qingfei You, Christoforos Nalmpantis, Edouard Grave, and Sebastian Riedel. Peer: A collaborative language model. *arXiv preprint arXiv:2208.11663*, 2022.

Thomas Scialom, Paul-Alexis Dray, Sylvain Lamprier, Benjamin Piwowarski, Jacopo Staiano, Alex Wang, and Patrick Gallinari. QuestEval: Summarization asks for fact-based evaluation. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6594–6604, Online and Punta Cana, Dominican Republic, November 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.emnlp-main.529. URL `https://aclanthology.org/2021.emnlp-main.529`.

Derek Tam, Anisha Mascarenhas, Shiyue Zhang, Sarah Kwan, Mohit Bansal, and Colin Raffel. Evaluating the factual consistency of large language models through news summarization. In Anna Rogers, Jordan Boyd-Graber, and Naoaki Okazaki, editors, *Findings of the Association for Computational Linguistics: ACL 2023*, pages 5220–5255, Toronto, Canada, July 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.findings-acl.322. URL `https://aclanthology.org/2023.findings-acl.322`.

Yi Tay, Vinh Tran, Mostafa Dehghani, Jianmo Ni, Dara Bahri, Harsh Mehta, Zhen Qin, Kai Hui, Zhe Zhao, Jai Gupta, et al. Transformer memory as a differentiable search index. *Advances in Neural Information Processing Systems*, 35:21831–21843, 2022.

James Thorne, Andreas Vlachos, Oana Cocarascu, Christos Christodoulopoulos, and Arpit Mittal. The fact extraction and VERification (FEVER) shared task. In *Proceedings of the First Workshop on Fact Extraction and VERification (FEVER)*, pages 1–9, Brussels, Belgium, November 2018. Association for Computational Linguistics. doi: 10.18653/v1/W18-5501. URL `https://aclanthology.org/W18-5501`.

Neeraj Varshney, Swaroop Mishra, and Chitta Baral. Investigating selective prediction approaches across several tasks in IID, OOD, and adversarial settings. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 1995–2002, Dublin, Ireland, May 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.findings-acl.158. URL `https://aclanthology.org/2022.findings-acl.158`.

Michael Völske, Martin Potthast, Shahbaz Syed, and Benno Stein. TL;DR: Mining Reddit to learn automatic summarization. In *Proceedings of the Workshop on New Frontiers in Summarization*, pages 59–63, Copenhagen, Denmark, September 2017. Association for Computational Linguistics. doi: 10.18653/v1/W17-4508. URL `https://aclanthology.org/W17-4508`.

Alex Wang, Kyunghyun Cho, and Mike Lewis. Asking and answering questions to evaluate the factual consistency of summaries. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5008–5020, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.450. URL https://aclanthology.org/2020.acl-main.450.

Sean Welleck, Jason Weston, Arthur Szlam, and Kyunghyun Cho. Dialogue natural language inference. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3731–3741, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1363. URL https://aclanthology.org/P19-1363.

Dustin Wright, David Wadden, Kyle Lo, Bailey Kuehl, Arman Cohan, Isabelle Augenstein, and Lucy Lu Wang. Generating scientific claims for zero-shot scientific fact checking. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2448–2460, Dublin, Ireland, May 2022. Association for Computational Linguistics. doi: 10.18653/v1/2022.acl-long.175. URL https://aclanthology.org/2022.acl-long.175.

Ori Yoran, Tomer Wolfson, Ben Bogin, Uri Katz, Daniel Deutch, and Jonathan Berant. Answering questions by meta-reasoning over multiple chains of thought. In Houda Bouamor, Juan Pino, and Kalika Bali, editors, *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 5942–5966, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.emnlp-main.364. URL https://aclanthology.org/2023.emnlp-main.364.

Hiyori Yoshikawa and Naoaki Okazaki. Selective-LAMA: Selective prediction for confidence-aware evaluation of language models. In Andreas Vlachos and Isabelle Augenstein, editors, *Findings of the Association for Computational Linguistics: EACL 2023*, pages 2017–2028, Dubrovnik, Croatia, May 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.findings-eacl.150. URL https://aclanthology.org/2023.findings-eacl.150.

Xiang Yue, Boshi Wang, Ziru Chen, Kai Zhang, Yu Su, and Huan Sun. Automatic evaluation of attribution by large language models. In Houda Bouamor, Juan Pino, and Kalika Bali, editors, *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 4615–4635, Singapore, December 2023. Association for Computational Linguistics. doi: 10.18653/v1/2023.findings-emnlp.307. URL https://aclanthology.org/2023.findings-emnlp.307.

Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. HellaSwag: Can a machine really finish your sentence? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4791–4800, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1472. URL https://aclanthology.org/P19-1472.

Hanning Zhang, Shizhe Diao, Yong Lin, Yi Fung, Qing Lian, Xingyao Wang, Yangyi Chen, Heng Ji, and Tong Zhang. R-tuning: Instructing large language models to say ‘I don‘t know’. In Kevin Duh, Helena Gomez, and Steven Bethard, editors, *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 7113–7139, Mexico City, Mexico, June 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.naacl-long.394. URL https://aclanthology.org/2024.naacl-long.394/.

A Impact

As discussed in Section 1, one of the main disadvantages of current LMs is their tendency to factually mislead the user by generating factual incorrect statements. Hence, the main impact of our work is to reduce such factual mistakes via our proposed method. Still, it is evident that this sort of approach can by no means completely eliminate hallucinations. It is important to stress that we propose a single method, not a system design for safe deployment of LLMs. In practice, we anticipate our method to be coupled with other checks and balances, forming a safe system.

Additionally, in this work, we use The Pile as a dataset to train models. The Pile is a web-crawled corpus, which likely harbors text reflecting various forms of biases. One impact of applying `IDK`-tuning is that the model may learn to answer in a biased way if this bias appears in its training data, while avoiding answers that rarely appear in its training data. This shows the need for more research on compiling high-quality training corpora.

B Computational Resources

For `IDK`-tuning of `Mistral-7B-v0.1`, we use Nvidia H100 or A100 GPUs depending on availability. For `IDK`-tuning `pythia-70m – 2.8B`, we use 1-4 Nvidia A6000 GPUs. For `IDK`-tuning of `bert-base-cased`, we use a single Nvidia A100 GPU.

C Questions Rephrasing

As mentioned in Section 3.2, for TriviaQA and PopQA, where the input is formed as a question, we reduce each of these input examples into a sentence completion task input, using GPT4. If we denote a random input question from one of these datasets by x, then our prompt to GPT4 is the following:

```
Please rephrase the following question as an input for a sentence
completion task. For example:

For the question: "Where was Michael Jackson born?", the sentence
should be: "Michael Jackson was born in".

For the question: "Who is Barack Obama's wife", the sentence
should be: "The wife of Barack Obama is".

For the question: "Where in England was Dame Judi Dench born?",
the sentence should be:
```

We found this prompt to be effective enough after manually testing it on a development set of a 45 examples.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: For all claims made in the abstract and introduction, we provide experimental results that back these claims.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: See Section 7.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory Assumptions and Proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [NA]

Justification: We do not include new theoretical results that warrant proofs.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental Result Reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: We provide the formulation of our objective in Section 2 and hyperparameters as well as further details to reproduce our trainings in Section 3.1.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [No]

Justification: We will publish all datasets, code, and model checkpoints with camera-ready version.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental Setting/Details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: We provide these details in Section 3.1. See also our answer to question 4.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment Statistical Significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [No]

   Justification: Our evaluations are done via prompting rather than fine-tuning (see Section 3.2), yielding no source of randomness to aggregate into error bars. Our large-scale continual training experiments are, unfortunately, too expensive to repeat multiple times with different random seeds.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).

- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments Compute Resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: We provide those details in APPENDIX (see Appendix B).

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code Of Ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: We carefully read the NeurIPS Code of Ethics document and made sure it's aligned with our work. One potential impact is discussed in Appendix A.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader Impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [Yes]

    Justification: We discuss the potential social impacts in Appendix A.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.

- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: We only continually pretrain already public models on up to 1B tokens. We believe that the resulting checkpoints do not warrant additional safeguards.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: In Section 3, we mention and cite each of the models, datasets, and training technuiqs we used in our work.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.

- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New Assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [Yes]

    Justification: Yes, we properly explained each of the new assets we introduced. Additionally, in Appendix C, we provide the complete prompt we used in order to create our closed-booked sentence completion dataset as discussed in Section 3.2.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and Research with Human Subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: No crowdsourcing experiments with human subjects were conducted.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional Review Board (IRB) Approvals or Equivalent for Research with Human Subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [NA]

    Justification: No study with human participants was conducted.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.

- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.