ADACAD: Adaptively Decoding to Balance Conflicts between Contextual and Parametric Knowledge

Anonymous ACL submission

Abstract

Knowledge conflict arises from discrepancies between information in the context of a large language model (LLM) and the knowledge stored in its parameters. This can hurt performance when using standard decoding techniques, which tend to ignore the context. Existing test-time contrastive methods seek to address this by comparing the LLM's output distribution with and without the context and adjust the model according to the contrast between them. However, we find that these methods frequently misjudge the degree of conflict and struggle to handle instances that vary in their amount of conflict, with static methods over-adjusting when conflict is absent. We propose a fine-grained, instance-level approach called ADACAD, which dynamically infers the weight of adjustment based on the degree of conflict, as measured by the Jensen-Shannon divergence between distributions representing contextual and parametric knowledge. Across four LLMs, six question-answering (QA) and three summarization datasets, we demonstrate that ADACAD consistently outperforms other decoding baselines with average QA accuracy gains of 14.21% (absolute) over a static contrastive baseline, and improves the factuality of summaries by 5.59 (AlignScore). Lastly, we show that while contrastive baselines hurt performance when conflict is absent, ADACAD mitigates these losses, making it more applicable to real-world datasets in which some examples have conflict and others do not.[1]

1 Introduction

Large language models (LLMs) encode vast amounts of information from pretraining in their parameters (Petroni et al., 2019; Roberts et al., 2020), giving them remarkable capabilities in knowledge-intensive NLP tasks. However, LLMs also hallucinate plausible but factually incorrect responses due to outdated knowledge (Lazaridou et al., 2021; Dhingra et al., 2022; Kasai et al., 2023), lesser-known facts (Mallen et al., 2023), and even misinformation in the pre-training corpus. A popular line of prior work aims to improve answers and reduce hallucination by augmenting LLMs' context with external knowledge, including knowledge from retrieved documents (Guu et al., 2020; Lewis et al., 2020), web search results (Nakano et al., 2022), and the outputs of tools (Schick et al., 2023). However, discrepancies between the added contextual knowledge and the model's pretrained parametric knowledge can cause *knowledge conflict*. In these cases, models often overlook the provided context and rely overly on the parametric knowledge (Longpre et al., 2021; Chen et al., 2022; Zhou et al., 2023; Wan et al., 2023). For example, in Fig. 1, the LLM's pretraining data (and thus its parametric knowledge) has a cutoff of September 2022, at which point France had hosted the Sum-

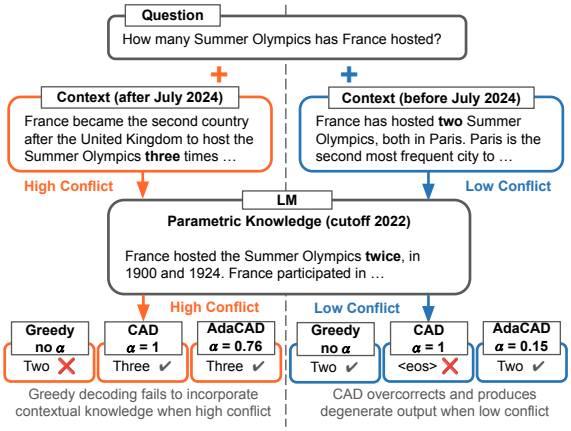

Figure 1: In cases of high knowledge conflict, greedy decoding fails to attend to the context, resulting in incorrect answers. Contrastive decoding allows the context to be incorporated, but must be done with care: in low-conflict cases, excessive contrast can over-correct (e.g., by CAD with $\alpha = 1$), resulting in incorrect outputs. ADACAD dynamically adjusts the degree of contrast, allowing it to handle both high and low-conflict cases.

[1]Code included in supplementary materials.

mer Olympics twice. This conflicts with the latest contextual knowledge (from July 2024) when France had hosted three times, and leads the model to answer incorrectly when using greedy decoding.

One promising direction for handling knowledge conflict uses inference-time decoding strategies that adjust the model's output probability distribution without the need for additional training. Shi et al. (2024) propose context-aware decoding (CAD) which seeks to correct the model's output based on the difference between output probability distributions with and without the context. However, in practice, we find that while CAD works well when there is a uniformly high degree of conflict between the parametric knowledge and external context, it struggles with scenarios in which different examples have *varying degrees of knowledge conflict*. Empirically, we observe that CAD can in fact degrade performance on low-conflict examples by *overcorrecting* the output distribution. For example, in Fig. 1, when the context is sourced from a document before July 2024, there is no conflict between the parametric knowledge and the contextual knowledge; both state that France has hosted the Olympics twice. Here, CAD overcorrects the distribution, leading to an invalid answer.

In this work, we present a simple and effective dynamic decoding method, **Ada**ptive **C**ontext **A**ware **D**ecoding (ADACAD) , aimed at automatically modeling the degree of conflict between the context and parametric knowledge and dynamically inferring the degree of adjustment needed for every token. We use the Jensen-Shannon divergence (JSD) between output distributions with and without the context to measure the degree of knowledge conflict, using the resulting value to reweight the combination of distributions. A higher JSD indicates a greater degree of conflict and signals the need for higher adjustment (more weight on the contextual knowledge) while a lower JSD reflects a smaller degree of conflict requiring a smaller adjustment (more weight on the parametric knowledge). As illustrated in Fig. 1, this leads to correct answers *both for high and low-conflict examples* by helping the model adaptively decide how to weigh contextual vs. parametric knowledge.

We demonstrate ADACAD's effectiveness on a diverse range of tasks, covering question-answering (QA) and summarization, with six QA datasets (Natural Question (NQ; Kwiatkowski et al., 2019), NQ-SWAP (Longpre et al., 2021), TriviaQA (Joshi et al., 2017), PopQA (Mallen et al., 2023), HotpotQA (Yang et al., 2018), TabMWP (Lu et al., 2023)) and three summarization datasets (CNN-DM (See et al., 2017), XSum (Narayan et al., 2018), and TofuEval (Tang et al., 2024)). We test a range of base LLMs, examining Llama-2 (Touvron et al., 2023), Llama-3 (AI@Meta, 2024), and Mistral (Jiang et al., 2023). We consider different sizes of these models and also test both the base and instruction-tuned variants. Our results and analyses show that decoding with a uniform level of contrast benefits high-conflict scenarios but generally hurts performance, while the adaptive contrast of ADACAD results in improvements across the board. Overall, ADACAD generally achieves superior performance compared to the baselines, with an absolute gain of 14.21% over CAD (a static baseline), 4.82% over COIECD (Yuan et al., 2024, a method that classifies instances as conflicting or not), and 2.41% over greedy decoding when averaged across models and QA datasets. On summarization, ADACAD improves summary quality and factuality, with an average AlignScore (Zha et al., 2023) gain of 4.16 over greedy decoding, 2.19 over CAD, and 10.44 over COIECD.

Furthermore, in our analyses, we explore *why* ADACAD improves over the baselines. We first validate the hypothesis that ADACAD is able to balance contextual and parametric knowledge by assigning lower weights to lower-conflict instances, testing each method on datasets designed to have high and low conflict, finding that ADACAD's inferred weight is much lower when there is no conflict. We also compare the amount by which CAD and ADACAD adjust the base model's distribution on examples with and without conflict, finding that while ADACAD changes the distribution less when there is no conflict (i.e., when the base model's distribution is already sufficient), CAD adjusts by roughly the same amount whether there is conflict or not, explaining its lower QA performance. Additionally, for summarization tasks, ADACAD generates more faithful summaries whereas other methods tend to hallucinate details.

2 Related Work

Knowledge Conflict. Integrating external knowledge as context into LLMs enables them to keep abreast of current world knowledge (Kasai et al., 2023), reduce hallucination (Shuster et al., 2021), and improve factuality. However, a recent line of work focuses on discrepancies between external

contextual knowledge and the model's parametric knowledge, such as LLMs' over-reliance on their parametric knowledge on entity-based QA tasks (Longpre et al., 2021), ignoring retrieved contexts (Tan et al., 2024), and exhibiting confirmation bias (Xie et al., 2024), etc. Zhou et al. (2023) demonstrate that LLMs' faithfulness to the context can be significantly improved using carefully designed prompting strategies – this is orthogonal to our work, which is compatible with different prompts. Zhang et al. (2023) address how to combine retrieved and parametric knowledge to improve open-domain QA, but require further training discriminators with silver labels, whereas our method is training-free.

Contrast in Text Generation. Contrastive approaches for text generation have been widely studied and used to enhance response diversity in conversations (Li et al., 2016), steering model generations towards desired attributes while maintaining fluency and diversity (Liu et al., 2021), and contrasting between larger and relatively smaller language models to generate high-quality text (Li et al., 2023), and improve visually-grounded generation tasks (Wan et al., 2024). Context-aware decoding (CAD; Shi et al., 2024) leverages a contrastive output distribution that amplifies the differences between the output probabilities predicted by a model with and without the context, promoting greater attention to the input context for more faithful and reliable text generation. Unlike AdaCAD, these past contrastive approaches do not adapt the weight on distributions to varying degrees of knowledge conflict. To address this, Yuan et al. (2024) introduce COIECD, a decoding-time method that categorizes instances into two discrete bins – high and low conflict – based on a complex information-entropy constraint governed by tuned hyperparameters, and employs different decoding strategies (by altering CAD) for each. In contrast, AdaCAD employs a single dynamic instance-level strategy that automatically models (based on Jensen-Shannon divergence) a *continuous degree of conflict without imposing rigid categories*. In addition to these conceptual differences, in Section 4.2, we show that AdaCAD outperforms both CAD and COIECD on QA and summarization.

3 Methodology

Task and Notation. Given an input query $\boldsymbol{x}$ with a relevant context $\boldsymbol{c}$, a language model parameterized by θ is tasked with generating a correct response $\boldsymbol{y} = y_1, \ldots, y_n$ of length n that respects the context. At each decoding step t, a token y_t can be sampled autoregressively from a probability distribution conditioned on query $\boldsymbol{x}$ and context $\boldsymbol{c}$ as $y \sim p_\theta(y \mid \boldsymbol{c}, \boldsymbol{x}, \boldsymbol{y}_{<t})$. However, when there is conflict between knowledge in the context $\boldsymbol{c}$ and parametric knowledge encoded in LLM, the model can struggle to pay enough attention to $\boldsymbol{c}$ and overly rely on the parametric knowledge (Longpre et al., 2021; Chen et al., 2022), i.e., sample from a distribution more akin to $p_\theta(y|\boldsymbol{x}, \boldsymbol{y}_{<t})$.

Background: Context-aware Decoding. To mitigate knowledge conflicts, Shi et al. (2024) introduce Context-aware Decoding (CAD), which samples from a contrastive output distribution that amplifies the difference between output probabilities with and without context. CAD measures the parametric knowledge via $p_\theta(y|\boldsymbol{x}, \boldsymbol{y}_t)$ and prioritizes relevant contextual knowledge over the model's parametric knowledge by using the pointwise mutual information (PMI) between the context $\boldsymbol{c}$ and the generation y, conditioned on $\boldsymbol{x}, \boldsymbol{y}_{<t}$ to modify the model's original output distribution.

$$\begin{aligned} y_t &\sim \tilde{p}_\theta\left(y \mid \boldsymbol{c}, \boldsymbol{x}, \boldsymbol{y}_{<t}\right) \\ &\propto p_\theta(y \mid \boldsymbol{c}, \boldsymbol{x}, \boldsymbol{y}_{<t}) \left[\frac{p_\theta(y \mid \boldsymbol{c}, \boldsymbol{x}, \boldsymbol{y}_{<t})}{p_\theta(y \mid \boldsymbol{x}, \boldsymbol{y}_{<t})}\right]^\alpha \end{aligned} \quad (1)$$

where the PMI term $\frac{p_\theta\left(y|\boldsymbol{c},\boldsymbol{x},\boldsymbol{y}_{<t}\right)}{p_\theta\left(y|\boldsymbol{x},\boldsymbol{y}_{<t}\right)}$ is a scaling factor used to adjust the parametric knowledge, and α governs the weight or degree of adjustment. A larger α means a greater adjustment and $\alpha = 0$ reduces to no adjustment, i.e., greedy decoding.[2]

AdaCAD: Handling Variable Conflict. In test-time contrastive methods (e.g., Li et al., 2023; Shi et al., 2024), α is a fixed hyperparameter set for an entire dataset, requiring tuning on a validation set. However, every instance in the dataset may need a different weight for adjustment; furthermore, in longer-form generation, individual timesteps may require different weights, making a single α value suboptimal. For instance, in the presence of a high degree of conflict, e.g., Fig. 2 (top), a larger α can perturb the LLM's output distribution to mitigate over-reliance on parametric knowledge, whereas in cases with low or no conflict (as in Fig. 2 (bottom)), the adjustment to the LLM's output distribution is

[2]While PMI also measures the amount of conflict between the distributions with and without context, empirically, we find that it still results in a high degree of perturbation to the output distribution in cases of low conflict (c.f. Section 5.2).

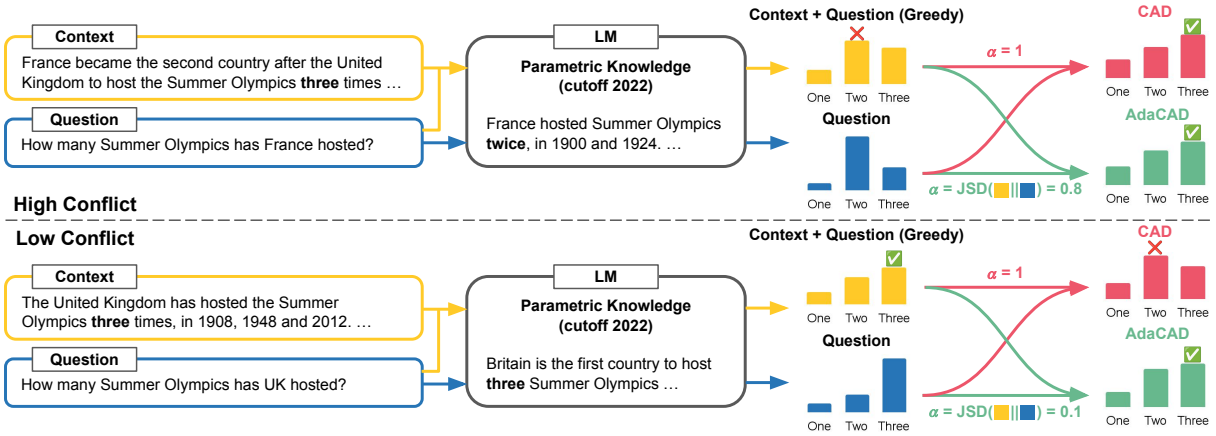

Figure 2: Comparison of greedy decoding (Context+Question), CAD, and ADACAD on high-conflict and low-conflict examples. Greedy decoding struggles to incorporate contextual knowledge in high-conflict examples. CAD tends to overemphasize irrelevant tokens in the vocabulary, leading to incorrect answers in low-conflict examples. ADACAD uses dynamic adaptation to effectively balance between context and parametric knowledge.

minimal. Therefore, a fixed α may fail on scenarios where there are heterogeneous examples with and without conflict, i.e., on realistic datasets.

To address variable conflict, we introduce a different α_t for each timestep and each instance. Specifically, we automatically infer α_t dynamically based on the degree of knowledge conflict for each instance (and decoding step) without supervision, enabling automatic adaptation. To accomplish this, we use Jensen-Shannon divergence (JSD; Lin, 1991) to model the degree of conflict between the context and parametric knowledge. While similar to Kullback-Leibler divergence, JSD is symmetric and bounded within the range $[0, 1]$, making it more suitable for modeling conflicts, as it provides a more interpretable and normalized measure of divergence (details in Appendix A). A larger JSD between $p_\theta(y \mid \boldsymbol{x}, \boldsymbol{y}_t)$ and $p_\theta(y \mid \boldsymbol{c}, \boldsymbol{x}, \boldsymbol{y}_t)$ reflects a greater conflict between context and parameter knowledge, suggesting that we need a larger α to encourage the LM to rely more on the context, while a smaller JSD reflects a smaller conflict, suggesting that a smaller α is required to maintain the LM's adherence to its parametric knowledge. Therefore, we set α_t^{JSD} at each decoding step t to:

$$\alpha_t^{\text{JSD}} = \text{JSD}\left(p_\theta\left(y_t \mid \boldsymbol{x}, \boldsymbol{y}_{<t}\right) \parallel p_\theta\left(y_t \mid \boldsymbol{c}, \boldsymbol{x}, \boldsymbol{y}_{<t}\right)\right)$$

This enables both coarse-grained instance-level and fine-grained token-level adjustments. Finally, we sample outputs from the probability distribution:

$$y_t \sim p_\theta(y \mid \boldsymbol{c}, \boldsymbol{x}, \boldsymbol{y}_{<t}) \left[\frac{p_\theta(y \mid \boldsymbol{c}, \boldsymbol{x}, \boldsymbol{y}_{<t})}{p_\theta(y \mid \boldsymbol{x}, \boldsymbol{y}_{<t})}\right]^{\alpha_t^{\text{JSD}}}$$

This dynamic adaptation allows our approach to effectively balance between context and parametric knowledge, ensuring robust performance across varying degrees of conflict without the need for extensive manual tuning, thereby enhancing both flexibility and accuracy in diverse scenarios.

ADACAD for Long-form Generation. In long-form generation tasks, we find that initially, the JSD values tend to be low (cf. Fig. 5 in Appendix A.1). This may be due to the model's tendency to produce generic, low-information outputs at the start of each sequence. Therefore, the divergence between $p_\theta(y_t|\boldsymbol{x}, \boldsymbol{y}_{<t})$ and $p_\theta(y_t|\boldsymbol{c}, \boldsymbol{x}, \boldsymbol{y}_{<t})$ is minimal. To mitigate this issue and ensure more consistent performance throughout the generation process, we introduce a warmup operation: $\alpha_t^{\text{JSD}} = \max\left(\alpha_t^{\text{JSD}}, \lambda\right)$, where λ is a lower bound to adjust for the initially low JSD values, ensuring a more robust and stable starting point. We set $\lambda = 0.3$ for long-form generation tasks.[3]

4 Experiments and Results

4.1 Experimental Setup

Datasets and Metrics. We evaluate on several QA datasets: Natural Questions (NQ; Kwiatkowski et al., 2019), TriviaQA (Joshi et al., 2017), PopQA (Mallen et al., 2023), and HotpotQA (Yang et al., 2018). We use these datasets to simulate real scenarios with varying degrees of conflict for each instance. Additionally, we evaluate on an existing knowledge conflict dataset, NQ-SWAP (Longpre et al., 2021), which is based on the NQ dataset and consists of synthetic conflicting data. Lastly, we also test on a popular tabular question-answering dataset, TabMWP (Lu et al., 2023), that requires

[3]We set $\lambda = 0.3$ to match the maximum JSD values for non-conflicting data from QA (cf. Section 5.1).

LLMs to use reasoning skills over tabular contexts. We report exact match accuracy on all QA datasets.

To test ADACAD on longer-form generation tasks, we evaluate on three standard summarization tasks: CNN-DM (See et al., 2017), XSum (Narayan et al., 2018), and TofuEval (Tang et al., 2024). While many documents from older datasets (such as CNN-DM and XSum) are present in LLM's pretraining data,[4] TofuEval is a recent, more challenging benchmark on topic-focused dialogue summarization (especially for marginal or secondary topics in the document). We use two reference-based metrics, ROUGE-L (Lin, 2004) and BERT-P (Zhang et al., 2020), to evaluate summarization quality. As TofuEval does not support reference-based evaluation Tang et al. (2024), we use recommended AlignScore (Zha et al., 2023) to measure the factual consistency of summaries on both *main* (central to the document) and *marginal* (lesser explored) topics. For additional details and examples of all datasets, refer to Appendix B.

Source of Context. We use the gold context provided by NQ, NQ-SWAP, TriviaQA, and HotpotQA as the relevant contexts. Since PopQA does not provide gold contexts, we employ BM25 (Robertson and Zaragoza, 2009), to retrieve relevant contexts from Wikipedia. For TabMWP, we take the semi-structured table as the relevant context. In summarization tasks, the source document serves as the relevant context, while the instruction is used as the input query. A summary of input query $\boldsymbol{x}$ and context $\boldsymbol{c}$ for all datasets is shown in Table 9 with corresponding prompts in Appendix F.

Models. We test ADACAD on different pretrained base language models, including Llama2 (13B) (Touvron et al., 2023), Llama3 (8B, 70B) (AI@Meta, 2024), and Mistral (7B) (Jiang et al., 2023); we measure ADACAD's effectiveness both on the base and instruction-tuned model variants.

Baselines. We compare ADACAD to standard decoding, context-aware decoding (CAD; Shi et al., 2024) – which has a fixed α, and COIECD (Yuan et al., 2024) – which classifies whether there is knowledge conflict using a method controlled by tuned thresholds and then operates in two different decoding modes, each with the same fixed α. Across all tasks and baselines, we use greedy decoding under a zero-shot setting.[5] For CAD, we set $\alpha = 1$ for the QA datasets and $\alpha = 0.5$ for the summarization datasets, following prior work (Shi et al., 2024). For COIECD, the values of λ and α are set to 0.25 and 1 for QA datasets, and 0.25 and 0.5 for the summarization datasets, respectively, following Yuan et al. (2024). In ADACAD, the α values are dynamically adjusted based on the degree of knowledge conflict for each instance.

4.2 Main Results

QA Tasks. From Table 1, we observe that ADACAD *consistently* outperforms greedy decoding, CAD, and COIECD. For instance, on Llama3-70B, ADACAD achieves an average score improvement of 2.18% (absolute) over greedy decoding, 12.91% over CAD, and 3.52% over COIECD. Note that while CAD performs quite well on NQ-SWAP (containing *only* high-conflict examples), it often degrades performance (relative to greedy decoding) on other QA datasets, resulting in an 18.58% accuracy drop on average across all models and tasks; in contrast, ADACAD performs well across datasets, whether they have conflict or not. Furthermore, ADACAD consistently outperforms COIECD across various QA datasets, highlighting the strength of our continuous JSD-based approach over COIECD's binary classification approach that splits instances into ones with conflict or without. For instance, ADACAD outperforms COIECD by a large average margin of 10.29% on NQ-SWAP across all models. Additionally, on more complex datasets like TabMWP with newer LLMs, ADACAD also shows superior performance against all baselines, e.g., achieving average improvements of 7.20% with Llama3-70B and 9.50% with Mistral-7B. These results indicate that ADACAD is better able to combine the advantages of greedy decoding and CAD, performing well in scenarios without knowledge conflict (as greedy decoding does) as well as those with conflict (as CAD does).

Summarization Tasks. In Table 2, we investigate how ADACAD can improve performance on longer-form generation, showing results on three summarization tasks, CNN-DM, Xsum, and TofuEval. For TofuEval, ADACAD demonstrates substantial improvements, particularly excelling in marginal topics (i.e., topics not central to the docu-

[4] Using pile.dataportraits.org, we find several documents from CNN-DM appear in the Pile (Gao et al., 2020), commonly used to pretrain LLMs.

[5] We find that greedy decoding outperforms top-p sampling on CNN-DM, so we use greedy decoding across all methods for summarization tasks instead of top-p sampling as in Shi et al. (2024) (c.f. Table 8, Appendix D).

Model	Decoding	NQ	NQ-SWAP	TriviaQA	PopQA	HotpotQA	TabMWP	Avg
Llama2-13B	Greedy	44.26	54.89	85.50	76.65	38.27	38.30	56.31
	CAD	37.91	**80.35**	71.40	76.83	31.92	19.30	52.95
	COIECD	44.60	59.84	**87.00**	**81.05**	**42.81**	**38.80**	**59.02**
	ADACAD	**46.73**	67.84	85.40	78.79	37.83	37.50	**59.02**
Llama3-8B	Greedy	44.63	47.81	**85.70**	80.51	**51.42**	52.20	60.38
	CAD	35.96	**77.94**	40.20	74.27	39.53	26.60	49.08
	COIECD	43.36	51.16	83.10	78.49	45.63	49.70	58.57
	ADACAD	**45.47**	62.34	82.50	**81.34**	50.53	**53.00**	**62.53**
Llama3-70B	Greedy	44.13	55.74	**90.20**	**86.10**	**56.11**	66.70	66.50
	CAD	34.05	**81.32**	54.60	75.16	40.86	48.60	55.77
	COIECD	45.09	57.26	88.60	83.60	52.03	64.40	65.16
	ADACAD	**45.43**	70.07	88.80	85.68	55.00	**67.10**	**68.68**
Mistral-7B	Greedy	42.56	56.86	**80.40**	**67.56**	40.89	38.90	57.65
	CAD	20.98	66.89	24.20	48.54	18.49	20.10	35.82
	COIECD	29.00	58.09	71.60	64.59	35.83	31.60	48.45
	ADACAD	**45.09**	**67.27**	80.20	67.26	**41.35**	**39.70**	**60.23**

Table 1: Under a zero-shot setting, we show that on average (across tasks and models) ADACAD improves accuracy by 14.21% over CAD and 4.82% over COIECD (results with instruction-tuned models in Appendix C).

	CNN-DM			XSum			TofuEval (AlignScore)	
Decoding	ROUGE-L	BERT-P	AlignScore	ROUGE-L	BERT-P	AlignScore	Overall	Main / Marginal
Greedy	24.93	**95.41**	91.44	14.36	94.05	85.28	76.66	81.64 / 61.19
CAD	24.76	94.45	91.01	14.59	93.65	84.34	83.93	87.26 / 73.58
COIECD	23.47	92.06	85.49	14.51	91.04	73.81	75.24	80.68 / 58.31
ADACAD	**25.42**	94.91	**94.97**	**14.91**	**94.29**	**85.81**	**85.07**	**88.06 / 75.79**

Table 2: Results on summarization datasets with Llama3-70B showing ADACAD yields the best performance on factuality metrics (AlignScore) and overall summarization quality (ROUGE-L, and BERT-P). The full results with other language models are shown in Table 7 of Appendix E.

ment) where it outperforms greedy decoding, CAD, and COIECD by 14.60, 2.21, and 17.48 points in terms of AlignScore – a measure of faithfulness – respectively. This highlights ADACAD's ability to handle diverse topics and maintain factual consistency, especially when prompted to focus on a marginal topic; qualitatively, we see in Fig. 4 that these improvements are driven by less hallucination on the part of ADACAD.

On CNN-DM, ADACAD achieves the highest ROUGE-L score of 25.42, surpassing greedy decoding, CAD, and COIECD by 0.49, 0.66, and 1.95, respectively. In terms of factual consistency, ADACAD also leads with an AlignScore of 94.97. On XSum, ADACAD also outperforms all baselines across all metrics. For instance, ADACAD achieves an average improvement of 1.38, and 4.67 points in BERT-P, and AlignScore, respectively. For BERT-P metric on CNN-DM, ADACAD outperforms CAD and COIECD and is slightly lower than Greedy decoding; as mentioned in Section 4.1, this may be a result of a lack of conflict in these datasets, which are at least partly included in large pretraining corpora. These improvements indicate that ADACAD's dynamic adjustment mechanism is effective in long-form generation, allowing it to balance context and parametric knowledge.

5 Analysis

5.1 Performance comparison on instances with higher and lower degrees of conflict

Setup. In Table 1, we find that CAD underperforms ADACAD, COIECD, as well as greedy decoding on most QA datasets, except NQ-SWAP, wherein every instance by design has a high degree of conflict (Longpre et al., 2021). We hypothesize that on more realistic datasets, the trailing performance of CAD stems from its inability to account for instances with low or minimal conflict. To test this hypothesis, we evaluate all methods on examples designed to have *minimal* conflict, i.e., where the model's internal representation aligns well with the context. Specifically, we generate a dataset of synthetic non-conflicting data called NQ-SYNTH: we sample 500 questions from Natural Questions and then prompt the Llama-3-70B to generate the answer for each question. We replace the gold answer entity in the context with the generated answer by

Decoding	NQ-SWAP	NQ-SYNTH	Overall
Greedy	51.60	88.20	69.90
CAD	79.60	64.00	71.80
COIECD	50.80	83.60	67.20
ADACAD	62.80	86.40	**74.60**

Table 3: Accuracy on conflicting data (NQ-SWAP) and non-conflicting data (NQ-SYNTH) with Llama3-70B.

regex, thus, making the context consistent with the LLM's internal knowledge. Finally, we evaluate Llama-3-70B on NQ-SYNTH and on NQ-SWAP.

Result: CAD hurts performance when conflict is low, while ADACAD can handle both cases. Consistent with our hypothesis, in Table 3, we observe that in the absence of conflict (on NQ-SYNTH), CAD substantially degrades performance by $\approx 24\%$ relative to greedy decoding, while ADACAD maintains a comparable performance. Although COIECD seeks to detect conflict and operates in two distinct decoding modes for high and low confllict, it also underperforms in non-conflict scenarios, falling 2.8% behind ADACAD. However, in cases of high conflict (NQ-SWAP), where greedy decoding yields dramatically lower accuracy, ADACAD improves over greedy decoding by 11.2%, while COIECD cannot handle high-conflict examples as well, lagging behind ADACAD by 12%. To further investigate how ADACAD balances instances with lower and higher degrees of conflict, we compute $\alpha^{\text{JSD}}_{\text{max}}$, which is the maximum α^{JSD}_t value across tokens, for both datasets. Indeed, we find that $\alpha^{\text{JSD}}_{\text{max}}$ adapts to the amount of conflict, with an average value of 0.45 on NQ-SWAP with a higher level of conflict, and substantially lower value ($\alpha^{\text{JSD}}_{\text{max}} = 0.28$) on NQ-SYNTH which does not contain any conflict by design.

5.2 PMI does not adequately address conflict

As described in Section 3, both CAD and ADACAD compute the PMI between the LLM's output distributions with and without external context $\boldsymbol{c}$. However, CAD relies solely on the PMI term to balance the level of conflict, whereas in ADACAD, we compute $(\text{PMI})^{\alpha^{\text{JSD}}_t}$ where both PMI and α^{JSD}_t adapt with the degree of conflict. In cases of low conflict, the LLM's distributions should in principle be the same with and without context, rendering PMI≈ 1, i.e., resorting to greedy decoding for any value of α (cf. Eq. (1)). However, in practice, we find that, even with minimal conflict, the PMI term reranks the tokens in the head of the LLM's distribution, resulting in poor performance for CAD.

Decoding	ρ (NQ-SWAP)	ρ (NQ-SYNTH)	$\lvert\Delta\rho\rvert$
CAD	0.56	0.57	0.01
ADACAD	0.86	0.94	0.08

Table 4: Spearman rank-order correlation coefficient between original and adjusted output distributions for CAD and ADACAD on NQ-SWAP and NQ-SYNTH. The difference $|\Delta\boldsymbol{\rho}|$ measures the sensitivity of a decoding method to the degree of conflict (higher is better).

Setup. To test how well the PMI term accounts for conflict, we measure the amount of reranking (among tokens) done by CAD and ADACAD relative to the greedy distribution. We compute the Spearman rank-order correlation coefficient ρ between the greedy distribution and output distribution from CAD and ADACAD (with scaling factors PMI and $(\text{PMI})^{\alpha^{\text{JSD}}_t}$ respectively). We restrict the measurement to the top-20 tokens (averaged across decoding steps) on NQ-SWAP and NQ-SYNTH.[6] Intuitively, a method *sensitive to the degree of conflict* should yield a lower rank correlation (more perturbation) when the amount of conflict is high (on NQ-SWAP), and higher rank correlation (less perturbation) in cases of low conflict (on NQ-SYNTH). To this end, we compute the absolute difference or *sensitivity*, $|\Delta\boldsymbol{\rho}|$ between the two $\boldsymbol{\rho}$ values of NQ-SWAP and NQ-SYNTH. A larger $|\Delta\boldsymbol{\rho}|$ indicates that the method is more effective at distinguishing between conflicting and non-conflicting data, i.e., more sensitive to the degree of conflict in instances.

Result: PMI over-perturbs greedy distribution in low conflict setting; ADACAD is adaptive. Results in Table 4 demonstrate that CAD, which only relies on the PMI term to offset conflicts, perturbs the greedy distribution to roughly the same extent (ρ) in the presence or absence of conflict, i.e., on NQ-SWAP and NQ-SYNTH, respectively. This minimal difference in $|\Delta\boldsymbol{\rho}|$ suggests that CAD is agnostic to the amount of conflict, leading to over-correction for non-conflicting examples. On the other hand, the correlation coefficient of ADACAD is higher on NQ-SYNTH than on NQ-SWAP (0.94 vs. 0.86), indicating more perturbation to the greedy distribution in the presence of conflict. Additionally, the sensitivity to conflict ($|\Delta\boldsymbol{\rho}|$) of ADACAD is substantially larger ($8\times$) than that of CAD, highlighting ADACAD's superior ability to distinguish between conflicting and non-conflicting examples. Note that ADACAD has a higher ρ in

[6] As the rank of low-probability tokens does not influence the generation, we focus on the top-20 tokens at each step.

Context (CN Tower Wikipedia Article):
A metal staircase reaches the main deck level after 1,776 steps, and the SkyPod 100 m (328 ft) above after 2,579 steps; ... but the fastest climb on record is 7 minutes and 52 seconds in 1989 by Brendan Keenoy, an Ontario Provincial Police officer. ... Paralympic champion Jeff Adams climbed the stairs of the tower in a specially designed wheelchair.
Question: *How many steps does the CN tower have?*
Gold Answer: 2,579 | **Parametric Knowledge (Llama3-70B):** 1776
Greedy: 1,776 steps **CAD:** Brendan Keenoy climbed the stairs in 7 minutes and 52 seconds in 1989. Jeff Adams climbed the stairs in a specially wheelchair. **COIECD:** 1,776 steps
AdaCAD: 2,579 steps

Figure 3: Qualitative example from NQ. Informative text is highlighted in blue, while text about unrelated facts and wrong answers is highlighted in red. ADACAD produces the correct answer, while CAD generates unrelated outputs and COIECD fails to detect the conflict and generates the same incorrect answer as greedy decoding.

Meeting Transcripts:
… they discussed the report on air quality impacts from Paramount's metal processing facilities on Long Beach residents …
Speaker 4: Katie, public comment CNN. Okay. We'll go ahead and do the this motion for this for this report. Then with consensus, no objection. Next item.
Speaker 6: To.
Speaker 1: Item 21 is a report from financial management recommendations received for the fiscal year 2016 year end budget performance report citywide.
Topic: Fiscal Year 2016 | Type: marginal
Greedy Decoding: The City Council received and filed the Fiscal Year 2016 Year-End Budget Performance Report. The report provides a summary of the City's financial performance for the fiscal year ended September 30, 2016.
CAD: Item 21 is a report from financial management recommendations received for the fiscal year 2016 year end budget performance report citywide.\nSpeaker 7: Mr. Mayor, members of the council, I'm Pat West, Long Beach City Manager. I'm joined by John Gross, our director of Financial Management. And we're here to present the fiscal year 2016 year end …
COIECD: The fiscal year 2016 year end budget performance report is a citywide report that provides a summary of the city's financial performance for the fiscal year 2016. The report includes information on the city's revenues, expenditures, and fund balances, as well as recommendations for improving financial performance. The report is intended to provide …
AdaCoD: The City Council received and filed the Fiscal Year 2016 Year-End Budget Performance Report Citywide.

Figure 4: TofuEval: Text unsupported by the transcripts is highlighted in red, while consistent and relevant text is highlighted in blue. ADACAD generates a faithful summary whereas other methods tend to hallucinate details.

both settings, indicating that overall, it perturbs the LLM's distribution to a lesser extent.

5.3 Qualitative Examples

QA Tasks. We highlight the importance of adaptively capturing the degree of conflict between the context and parametric knowledge, we show a qualitative example from the NQ dataset in Fig. 3. ADACAD outperforms other methods by correctly generating the "*CN Tower's 2,579 steps*". We also observe that CAD tends to produce unrelated outputs due to over-correction, which over-amplifies the influence of irrelevant tokens within the vocabulary. COIECD fails to detect the conflict and generates the same incorrect answer as greedy decoding. We find that ADACAD strikes the best balance between providing contrast in scenarios with high knowledge conflict while not suffering from over-correction on low-conflict instances.

Summarization. We also show a qualitative example from TofuEval in Fig. 4. Given a meeting transcript centered on *"report on air quality impacts from Paramount's metal processing facilities on Long Beach residents"*, we prompt the LLMs to generate a summary focused on the marginal topic *"Fiscal Year 2016"*. Baselines like greedy decoding, CAD, and COIECD tend to hallucinate details, such as fabricated financial data or names of individuals not mentioned in the transcript, which are highlighted in red. In contrast, ADACAD generates a more accurate and faithful summary without introducing unverified information.

6 Discussion and Conclusion

In naturalistic scenarios with mixed datasets containing examples with and without knowledge conflicts, existing decoding methods, including CAD, fail to adapt to changing amounts of conflict and in fact can lead to *reduced* performance. Although larger and more performant models can store more information in their parametric knowledge – thus leading to less and less conflict as models improve – there will still always be gaps between the model and the actual state of the world (e.g., because of time cutoffs). This means that models will encounter both low- and high-conflict scenarios, no matter their strength.

To this end, we introduce ADACAD, a simple yet effective dynamic decoding method that uses Jensen-Shannon divergence to dynamically model the degree of conflict for a given example (and timestep) and automatically balance the contrast between contextual and parametric knowledge. On diverse QA datasets, we show that ADACAD combines the best of greedy decoding and context-aware decoding, improving performance. Additionally, experiments on summarization demonstrate that ADACAD enhances both the quality and factuality of generated text, while other methods tend to hallucinate details. Lastly, ADACAD consistently outperforms COIECD, another hybrid decoding strategies that detects conflict. Our analysis reveals that ADACAD mitigates the overcorrection seen in CAD by dynamically adjusting the weight of contextual knowledge based on the degree of conflict.

Limitations

Since our proposed method ADACAD is based on CAD, it requires access to output logits from LLMs to calculate the difference between output probabilities with and without context. However, API-based LLMs like GPT-4 often do not provide output logits, making it challenging to directly apply logit-based methods like ADACAD and CAD to fully black-box models. Additionally, our experiments focus on English datasets and pre-trained models; as LLMs become available for other languages, future research will be needed to explore the interactions between language and knowledge conflict. We do not foresee any particular risks associated with the application of our method.

References

AI@Meta. 2024. Llama 3 model card.

Hung-Ting Chen, Michael Zhang, and Eunsol Choi. 2022. Rich knowledge sources bring complex knowledge conflicts: Recalibrating models to reflect conflicting evidence. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 2292–2307, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Bhuwan Dhingra, Jeremy R. Cole, Julian Martin Eisenschlos, Daniel Gillick, Jacob Eisenstein, and William W. Cohen. 2022. Time-aware language models as temporal knowledge bases. *Transactions of the Association for Computational Linguistics*, 10:257–273.

Leo Gao, Stella Biderman, Sid Black, Laurence Golding, Travis Hoppe, Charles Foster, Jason Phang, Horace He, Anish Thite, Noa Nabeshima, et al. 2020. The pile: An 800gb dataset of diverse text for language modeling. *arXiv preprint arXiv:2101.00027*.

Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, and Mingwei Chang. 2020. Retrieval augmented language model pre-training. In *Proceedings of the 37th International Conference on Machine Learning*.

Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi. 2020. The curious case of neural text degeneration. In *International Conference on Learning Representations*.

Yebowen Hu, Timothy Ganter, Hanieh Deilamsalehy, Franck Dernoncourt, Hassan Foroosh, and Fei Liu. 2023. MeetingBank: A benchmark dataset for meeting summarization. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 16409–16423, Toronto, Canada. Association for Computational Linguistics.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2023. Mistral 7b. *arXiv preprint arXiv:2310.06825*.

Mandar Joshi, Eunsol Choi, Daniel Weld, and Luke Zettlemoyer. 2017. TriviaQA: A large scale distantly supervised challenge dataset for reading comprehension. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1601–1611, Vancouver, Canada. Association for Computational Linguistics.

Jungo Kasai, Keisuke Sakaguchi, yoichi takahashi, Ronan Le Bras, Akari Asai, Xinyan Velocity Yu, Dragomir Radev, Noah A. Smith, Yejin Choi, and Kentaro Inui. 2023. Realtime QA: What's the answer right now? In *Thirty-seventh Conference on Neural Information Processing Systems Datasets and Benchmarks Track*.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, Kristina Toutanova, Llion Jones, Matthew Kelcey, Ming-Wei Chang, Andrew M. Dai, Jakob Uszkoreit, Quoc Le, and Slav Petrov. 2019. Natural questions: A benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:452–466.

Angeliki Lazaridou, Adhiguna Kuncoro, Elena Gribovskaya, Devang Agrawal, Adam Liska, Tayfun Terzi, Mai Gimenez, Cyprien de Masson d'Autume, Tomáš Kočiský, Sebastian Ruder, Dani Yogatama, Kris Cao, Susannah Young, and Phil Blunsom. 2021. Mind the gap: Assessing temporal generalization in neural language models. In *Advances in Neural Information Processing Systems*.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. In *Advances in Neural Information Processing Systems*.

Jiwei Li, Michel Galley, Chris Brockett, Jianfeng Gao, and Bill Dolan. 2016. A diversity-promoting objective function for neural conversation models. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 110–119, San Diego, California. Association for Computational Linguistics.

Xiang Lisa Li, Ari Holtzman, Daniel Fried, Percy Liang, Jason Eisner, Tatsunori Hashimoto, Luke Zettlemoyer, and Mike Lewis. 2023. Contrastive decoding: Open-ended text generation as optimization. In

Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 12286–12312, Toronto, Canada. Association for Computational Linguistics.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Jianhua Lin. 1991. Divergence measures based on the shannon entropy. *IEEE Transactions on Information theory*, 37(1):145–151.

Alisa Liu, Maarten Sap, Ximing Lu, Swabha Swayamdipta, Chandra Bhagavatula, Noah A. Smith, and Yejin Choi. 2021. DExperts: Decoding-time controlled text generation with experts and anti-experts. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 6691–6706, Online. Association for Computational Linguistics.

Shayne Longpre, Kartik Perisetla, Anthony Chen, Nikhil Ramesh, Chris DuBois, and Sameer Singh. 2021. Entity-based knowledge conflicts in question answering. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7052–7063, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Pan Lu, Liang Qiu, Kai-Wei Chang, Ying Nian Wu, Song-Chun Zhu, Tanmay Rajpurohit, Peter Clark, and Ashwin Kalyan. 2023. Dynamic prompt learning via policy gradient for semi-structured mathematical reasoning. In *International Conference on Learning Representations (ICLR)*.

Alex Mallen, Akari Asai, Victor Zhong, Rajarshi Das, Daniel Khashabi, and Hannaneh Hajishirzi. 2023. When not to trust language models: Investigating effectiveness of parametric and non-parametric memories. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 9802–9822, Toronto, Canada. Association for Computational Linguistics.

Reiichiro Nakano, Jacob Hilton, Suchir Balaji, Jeff Wu, Long Ouyang, Christina Kim, Christopher Hesse, Shantanu Jain, Vineet Kosaraju, William Saunders, Xu Jiang, Karl Cobbe, Tyna Eloundou, Gretchen Krueger, Kevin Button, Matthew Knight, Benjamin Chess, and John Schulman. 2022. Webgpt: Browser-assisted question-answering with human feedback. *arXiv preprint arXiv:2112.09332*.

Shashi Narayan, Shay B. Cohen, and Mirella Lapata. 2018. Don't give me the details, just the summary! topic-aware convolutional neural networks for extreme summarization. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1797–1807, Brussels, Belgium. Association for Computational Linguistics.

Fabio Petroni, Tim Rocktäschel, Sebastian Riedel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, and Alexander Miller. 2019. Language models as knowledge bases? In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2463–2473, Hong Kong, China. Association for Computational Linguistics.

Adam Roberts, Colin Raffel, and Noam Shazeer. 2020. How much knowledge can you pack into the parameters of a language model? In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5418–5426, Online. Association for Computational Linguistics.

Stephen E. Robertson and Hugo Zaragoza. 2009. The probabilistic relevance framework: Bm25 and beyond. *Foundations and Trends in Information Retrieval*.

Timo Schick, Jane Dwivedi-Yu, Roberto Dessi, Roberta Raileanu, Maria Lomeli, Eric Hambro, Luke Zettlemoyer, Nicola Cancedda, and Thomas Scialom. 2023. Toolformer: Language models can teach themselves to use tools. In *Thirty-seventh Conference on Neural Information Processing Systems*.

Abigail See, Peter J. Liu, and Christopher D. Manning. 2017. Get to the point: Summarization with pointer-generator networks. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1073–1083, Vancouver, Canada. Association for Computational Linguistics.

Weijia Shi, Xiaochuang Han, Mike Lewis, Yulia Tsvetkov, Luke Zettlemoyer, and Wen-tau Yih. 2024. Trusting your evidence: Hallucinate less with context-aware decoding. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 2: Short Papers)*, pages 783–791, Mexico City, Mexico. Association for Computational Linguistics.

Kurt Shuster, Spencer Poff, Moya Chen, Douwe Kiela, and Jason Weston. 2021. Retrieval augmentation reduces hallucination in conversation. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 3784–3803, Punta Cana, Dominican Republic. Association for Computational Linguistics.

Hexiang Tan, Fei Sun, Wanli Yang, Yuanzhuo Wang, Qi Cao, and Xueqi Cheng. 2024. Blinded by generated contexts: How language models merge generated and retrieved contexts when knowledge conflicts? *arXiv preprint arXiv:2401.11911*.

Liyan Tang, Igor Shalyminov, Amy Wong, Jon Burnsky, Jake Vincent, Yu'an Yang, Siffi Singh, Song Feng, Hwanjun Song, Hang Su, Lijia Sun, Yi Zhang, Saab Mansour, and Kathleen McKeown. 2024. TofuEval:

Evaluating hallucinations of LLMs on topic-focused dialogue summarization. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 4455–4480, Mexico City, Mexico. Association for Computational Linguistics.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

David Wan, Jaemin Cho, Elias Stengel-Eskin, and Mohit Bansal. 2024. Contrastive region guidance: Improving grounding in vision-language models without training. *arXiv preprint arXiv:2403.02325*.

David Wan, Shiyue Zhang, and Mohit Bansal. 2023. HistAlign: Improving context dependency in language generation by aligning with history. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 2941–2960, Singapore. Association for Computational Linguistics.

Jian Xie, Kai Zhang, Jiangjie Chen, Renze Lou, and Yu Su. 2024. Adaptive chameleon or stubborn sloth: Revealing the behavior of large language models in knowledge conflicts. In *The Twelfth International Conference on Learning Representations*.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William Cohen, Ruslan Salakhutdinov, and Christopher D. Manning. 2018. HotpotQA: A dataset for diverse, explainable multi-hop question answering. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2369–2380, Brussels, Belgium. Association for Computational Linguistics.

Xiaowei Yuan, Zhao Yang, Yequan Wang, Shengping Liu, Jun Zhao, and Kang Liu. 2024. Discerning and resolving knowledge conflicts through adaptive decoding with contextual information-entropy constraint. In *Findings of the Association for Computational Linguistics ACL 2024*, pages 3903–3922, Bangkok, Thailand and virtual meeting. Association for Computational Linguistics.

Yuheng Zha, Yichi Yang, Ruichen Li, and Zhiting Hu. 2023. AlignScore: Evaluating factual consistency with a unified alignment function. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11328–11348, Toronto, Canada. Association for Computational Linguistics.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020. Bertscore: Evaluating text generation with bert. In *International Conference on Learning Representations*.

Yunxiang Zhang, Muhammad Khalifa, Lajanugen Logeswaran, Moontae Lee, Honglak Lee, and Lu Wang. 2023. Merging generated and retrieved knowledge for open-domain QA. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 4710–4728, Singapore. Association for Computational Linguistics.

Wenxuan Zhou, Sheng Zhang, Hoifung Poon, and Muhao Chen. 2023. Context-faithful prompting for large language models. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 14544–14556, Singapore. Association for Computational Linguistics.

Chenguang Zhu, Yang Liu, Jie Mei, and Michael Zeng. 2021. MediaSum: A large-scale media interview dataset for dialogue summarization. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5927–5934, Online. Association for Computational Linguistics.

A Jensen-Shannon Divergence

Jensen–Shannon divergence (JSD) is a symmetric measure of the similarity between two probability distributions, defined as the average of the Kullback–Leibler divergences from their mean distribution. JSD between two probability distribution P and Q is defined as:

$$\mathrm{JSD}(P \parallel Q) = \frac{1}{2}\left(\mathrm{KL}(P \parallel M) + \mathrm{KL}(Q \parallel M)\right)$$

where $M = \frac{1}{2}(P + Q)$ is a mixture distribution of P and Q and:

$$\begin{aligned} \mathrm{KL}(P \parallel M) &= \textstyle\sum_x P(x) \log \frac{P(x)}{M(x)} \\ \mathrm{KL}(Q \parallel M) &= \textstyle\sum_x Q(x) \log \frac{Q(x)}{M(x)} \end{aligned}$$

A.1 JSD Value Trend for Summarization

Fig. 5 illustrates the trend of JSD values over the initial decoding steps when using LLama3-70B on

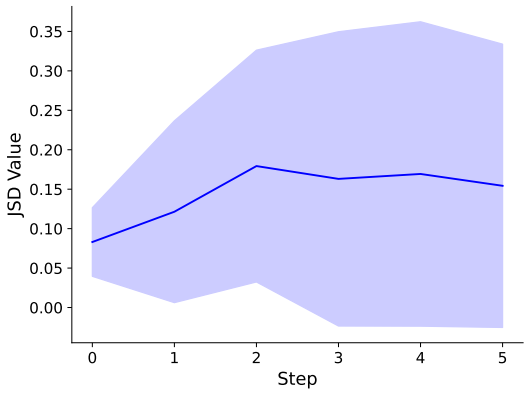

Figure 5: Plot of JSD values of the first 5 decoding steps using Llama3-70B on TofuEval. The JSD values tend to have lower values and variance at the start of decoding.

the TofuEval dataset. We observe that JSD values start relatively low and exhibit less variation or sensitivity in the early steps of decoding. This may be due to the model's tendency to produce generic, low-information outputs at the start of each sequence. As the decoding progresses, the JSD values increase and become more sensitive, indicating the dynamic adjustment in ADACAD works well.

B Dataset Details

We use six question answering datasets and three summarization datasets for evaluation. We also present one example from each dataset, as detailed in Table 9. For the synthetically generated QA datasets NQ-SWAP and NQ-SYNTH, we provide examples in Table 5.

B.1 QA Datasets

Some QA datasets, such as NQ, TriviaQA, and HotpotQA, do not have public test sets. For these we report performance of baselines and ADACAD on the dev set. Further, following Shi et al. (2024), to expedite inference, we sub-sample datasets where the test sets are very large (>8K instances).

- **Natural Question** (**NQ**; Kwiatkowski et al., 2019) is a large-scale QA dataset consisting of real user questions issued to Google search, with answers found from Wikipedia. We test on 3231 instances from the NQ validation set, which originally contained 7.83K examples. These instances were selected because they have short answers, making them suitable for evaluating all baselines and our method.

- **NQ-SWAP** (Longpre et al., 2021) introduces synthetic conflicts by swapping entities in the context to challenge the model's ability to manage conflicting information. Specifically, Longpre et al. (2021) first identify instances with named entity answers, then substitute mentions of the entity in the gold document with an alternate entity. NQ-SWAP consists of 4K instances derived from the NQ dataset.

- **TriviaQA** (Joshi et al., 2017) is a realistic QA dataset that includes a wide variety of trivia questions, requiring models to deal with large amounts of text from various sources and handle inference over multiple sentences. We randomly sample 1K instances from the TriviaQA Wiki validation set, which contains a total of 8K examples.

- **PopQA** (Mallen et al., 2023) is a dataset designed to test models' performance on questions about long-tail entities. We choose 1.6K instances from the PopQA test set for which we are successfully able to retireive contexts containing the gold answer (c.f. Section 4.1).

- **HotpotQA** (Yang et al., 2018) is a QA dataset that requires multi-hop reasoning, where the model needs to find and combine information from multiple sources to answer complex questions. We use the entire development set of HotpotQA, consisting of 7.4K instances.

- **TabMWP** (Lu et al., 2023) is a dataset focused on open-domain grade-level problems that require mathematical reasoning on both textual and tabular data. We use an official "lite" subset of TabMWP called "test1k" which contains 1K instances.

B.2 Summarization Datasets

- **CNN-DM** (See et al., 2017) is a widely used dataset for training and evaluating models on abstractive summarization tasks, involving news articles and their summaries. We randomly sample 500 examples from the original 11.5k test set.

- **XSum** (Narayan et al., 2018) is an abstractive summarization dataset known for its highly challenging nature, where the goal is to generate concise, one-sentence summaries from longer documents. We used 500 instances from the XSum dataset's 11.3K test set.

Question:
How many episodes are in Chicago Fire season 4?
NATURAL QUESTION
Original Context:
The fourth season of Chicago Fire contained 23 episodes. It is an American drama television series with ...
Original Answer: 23
NQ-SWAP
Substitute Context:
The fourth season of Chicago Fire contained 10 episodes. It is an American drama television series with ...
Substitute Answer: 10
NQ-SYNTH
Substitute Context:
The fourth season of Chicago Fire contained 22 episodes. It is an American drama television series with ...
Substitute Answer (generated from LLM): 22

Table 5: Example from NQ-SWAP and NQ-SYNTH. A substitute example for NQ-SWAP is made from the original example by replacing the original answer, 23, with a similar but conflicting answer, i.e., 10. A substitute example for NQ-SYNTH is made from the original example by replacing the original answer, 23, with one generated by Llama3-70B without context, i.e., 22.

- **TofuEval** (Tang et al., 2024) is a benchmark for evaluating the factual consistency and topic relevance of summaries, especially in scenarios involving dialogue or meeting transcriptions. This benchmark draws 50 test set documents from each of two datasets: MediaSum (Zhu et al., 2021) and MeetingBank (Hu et al., 2023). For each document, three topics were generated, resulting in a total of 300 topic-focused summaries. Approximately 75% of the total are main topics that refer to the central information in a document that is under discussion or is presented in the document, and the rest are marginal topics that refer to information in a document that is not the main focus of the document but is still part of the context.

B.3 Licenses

Datasets are released under the following licenses:

- Natural Questions: Apache-2.0 license
- NQ-Swap: MIT license
- TriviaQA: Apache-2.0 license
- PopQA: MIT license
- HotPotQA: Apache-2.0 license
- TabMWP: MIT license
- CNN-DM: Apache-2.0 license
- XSum: MIT license
- TofuEval: MIT license

The models we use have the following licenses:

- Llama 2: custom license `https://ai.meta.com/llama/license/`
- Llama 3: custom license `https://www.llama.com/llama3/license/`
- Mistral: Apache-2.0 license

C Instruction-tuned LLMs Experiments

We compare ADACAD against the baselines on all datasets using instruction-tuned language models and show the results in Table 6. We find that ADACAD achieves comparable or better performance than all baselines when applied to instruction-tuned models.

D Results of Different Decoding Methods on CNN-DM

Table 8 shows the results of different base decoding methods on CNN-DM with Llama-70B. Here, we see that greedy decoding performs better than Top-p sampling (Holtzman et al., 2020), motivating our use of greedy decoding in Table 2.

Decoding	ROUGE-L	BERT-P
Top-p Sampling	17.48	86.79
Greedy Decoding	**23.47**	**92.06**

Table 8: Comparison of greedy decoding and top-p sampling ($p = 0.9$) with Llama3-70B on CNN-DM.

E Full Results with Different Base LMs on Summarization Tasks

Table 7 shows the full results with all base language models on three summarization tasks: CNN-DM, XSum, and TofuEval. ADACAD achieves comparable or better performance than all baselines across different LLMs.

F Prompts

We provide the prompts for pre-trained base language with and without context for both QA and summarization tasks.

Model	Decoding	NQ	NQ-SWAP	TriviaQA	PopQA	HotpotQA	TabMWP	Avg
Llama2-13B-Chat	Greedy	35.75	50.24	54.40	**72.61**	32.15	50.40	49.26
	CAD	**39.49**	**71.24**	59.40	68.81	30.14	48.70	**52.96**
	ADACAD	37.08	57.69	**61.20**	72.31	**32.34**	**52.10**	52.12
Llama3-8B-Inst	Greedy	**40.27**	60.89	**64.00**	**70.89**	**39.66**	**68.50**	57.37
	CAD	39.43	**71.19**	52.30	70.35	37.27	63.10	55.61
	ADACAD	39.65	67.37	61.50	70.41	39.43	66.10	**57.41**
Llama3-70B-Inst	Greedy	40.82	59.16	64.10	64.41	47.70	70.40	57.77
	CAD	**42.31**	**66.37**	58.40	64.23	47.21	69.30	57.97
	ADACAD	41.35	60.77	**64.60**	**65.78**	**48.21**	**71.90**	**58.77**
Mistral-7B-Inst	Greedy	**42.93**	64.74	**77.20**	76.59	**50.26**	**50.20**	**60.32**
	CAD	42.56	**67.89**	71.70	74.45	47.12	46.40	58.35
	ADACAD	42.87	63.99	75.40	**76.89**	49.49	47.30	59.32

Table 6: Results on QA datasets with different instruction-tuned language models. When averaged across datasets, ADACAD is better than or comparable to the baselines.

Question Answering

With Context:
`{context}`
Using only the references listed above, answer the following question:
Question: `{question}`
Answer:

Without Context:
Answer the following question:
Question: `{question}`
Answer:

Summarization - CNN-DM

With Context:
Document: `{document}`
Summarize the document in three sentences.
Summary:

Without Context:
Summarize the document in three sentences.
Summary:

Summarization - TofuEval

With Context:
Document: `{document}`
Summarize the provided document focusing on "`{topic}`". The summary should be less than 50 words in length.
Summary:

Without Context:
Summarize the provided document focusing on "`{topic}`". The summary should be less than 50 words in length.
Summary:

Summarization - XSum

With Context:
Document: `{document}`
Summarize the document in one sentence.
Summary:

Without Context:
Summarize the document in one sentence.
Summary:

Decoding	CNN-DM			XSum			TofuEval (AlignScore)	
	ROUGE-L	**BERT-P**	**AlignScore**	**ROUGE-L**	**BERT-P**	**AlignScore**	**Overall**	**Main / Marginal**
				Llama2-13B				
Greedy	23.70	94.25	87.28	13.51	93.30	**85.23**	66.11	72.51 / 46.23
CAD	**24.33**	94.44	88.99	**14.86**	93.36	82.41	**80.39**	**84.03** / 69.07
COIECD	20.21	88.63	75.72	13.95	89.80	70.41	62.88	68.45 / 45.55
ADACAD	23.93	**94.63**	**91.15**	14.18	**94.04**	84.33	**80.39**	83.94 / **69.36**
				Llama3-8B				
Greedy	25.16	94.92	90.33	13.16	93.43	83.65	68.17	73.51 / 51.57
CAD	24.91	94.70	91.44	13.80	93.37	**86.88**	**83.40**	**86.77 / 72.94**
COIECD	23.60	92.01	83.92	13.65	91.40	69.47	70.07	73.65 / 58.94
ADACAD	**25.42**	**95.09**	**94.35**	**13.83**	**94.02**	86.78	80.62	83.24 / 72.46
				Llama3-70B				
Greedy	24.93	**95.41**	91.44	14.36	94.05	85.28	76.66	81.64 / 61.19
CAD	24.76	94.45	91.01	14.59	93.65	84.34	83.93	87.26 / 73.58
COIECD	23.47	92.06	85.49	14.51	91.04	73.81	75.24	80.68 / 58.31
ADACAD	**25.42**	94.91	**94.97**	**14.91**	**94.29**	**85.81**	**85.07**	**88.06 / 75.79**
				Mistral-7B				
Greedy	24.59	93.57	80.80	14.07	88.56	58.76	63.07	68.62 / 45.79
CAD	23.72	93.22	90.61	18.20	91.54	84.94	67.64	67.55 / **67.48**
COIECD	23.50	92.06	83.97	17.85	89.79	69.26	65.95	70.63 / 51.39
ADACAD	**24.76**	**94.21**	**93.05**	**18.51**	**92.19**	**86.79**	**74.00**	**77.59** / 62.84
				Llama3-70B-Instruct				
Greedy	24.72	90.64	88.22	**23.19**	90.80	82.40	78.56	80.18 / 73.52
CAD	25.17	**91.19**	88.52	20.92	**91.52**	**86.54**	79.86	79.55 / **80.82**
COIECD	23.85	89.84	83.88	22.41	90.61	81.42	77.54	78.69 / 73.97
AdaCAD	**25.26**	90.91	**88.68**	21.52	91.30	85.30	**81.16**	**82.82** / 76.03

Table 7: Results on summarization datasets with different LMs. ADACAD generally outperforms the baselines across metrics and datasets.

Natural Question
$\boldsymbol{c}$: The second season of the American television drama series Breaking Bad premiered on March 8, 2009 and concluded on May 31, 2009. It consisted of 13 episodes, each running approximately 47 minutes in length ... $\boldsymbol{x}$: How many episodes in season 2 Breaking Bad?
NQ-SWAP
$\boldsymbol{c}$: The second season of the American television drama series Breaking Bad premiered on March 8, 2009 and concluded on May 31, 2009. It consisted of 27 episodes, each running approximately 47 minutes in length ... $\boldsymbol{x}$: How many episodes in season 2 Breaking Bad?
TriviaQA
$\boldsymbol{c}$: ... Removal of dental biofilm is important as it may become acidic causing demineralization of the teeth (also known as caries) or harden into calculus (dental) (also known as tartar). Calculus can not be removed through ... $\boldsymbol{x}$: In dentistry, what is the name given to hardened dental plaque?
PopQA
$\boldsymbol{c}$: The 2012 Uzbekistan First League was the 21st season of 2nd level football in Uzbekistan since 1992. It is split in an Eastern and Western zone, each featuring 12 teams ... $\boldsymbol{x}$: What sport does 2012 Uzbekistan First League play?
HotpotQA
$\boldsymbol{c}$: <t> Superdrag </t> Superdrag was an American alternative rock band from Knoxville, Tennessee ... <t> Collective Soul </t> Collective Soul is an American rock band originally from Stockbridge, Georgia ... $\boldsymbol{x}$: Are both Superdrag and Collective Soul rock bands?
TabMWP
$\boldsymbol{c}$: alpaca | $1,605.00 kinkajou | $1,837.00 python | $8,343.00 parrot | $1,123.00 macaw | $1,629.00 $\boldsymbol{x}$: Erik has $7,616.00. How much money will Erik have left if he buys a parrot and a kinkajou? (Unit: $)
CNN-DM
$\boldsymbol{c}$: Article: (CNN)Two years ago, the storied Boston Marathon ended in terror and altered the lives of runners, spectators and those who tried to come to their rescue. Just last week, Dzhokhar Tsarnaev was convicted ... $\boldsymbol{x}$: Summarize the article in three sentences. Summary:
XSum
$\boldsymbol{c}$: You may want to choose another fantasy destination after the British Foreign Office told tourists to be aware that some political demonstrations in the capital, Male, have led to violence. It did add, though, that most trips ... $\boldsymbol{x}$: Summarize the article in one sentence. Summary:
TofuEval
$\boldsymbol{c}$: Document: DOBBS: General Motors today announced it will offer early retirement buyouts for 113,000 of its employees. Management calls it, "accelerated attrition". And it is only the latest sign of the dramatic decline ... $\boldsymbol{x}$: Summarize the provided document focusing on "Buyouts for General Motors employees". The summary should be less than 50 words in length. Summary:

Table 9: An illustration of input query $\boldsymbol{x}$ and relevant context $\boldsymbol{c}$ for different datasets.