# Synchronous Faithfulness Monitoring for Trustworthy Retrieval-Augmented Generation

**Di Wu, Jia-Chen Gu, Fan Yin, Nanyun Peng, Kai-Wei Chang**
University of California, Los Angeles
{diwu,fanyin20,violetpeng,kwchang}@cs.ucla.edu, gujc@ucla.edu

## Abstract

Retrieval-augmented language models (RALMs) have shown strong performance and wide applicability in knowledge-intensive tasks. However, there are significant trustworthiness concerns as RALMs are prone to generating unfaithful outputs, including baseless information or contradictions with the retrieved context. This paper proposes SynCheck, a lightweight monitor that leverages fine-grained decoding dynamics including sequence likelihood, uncertainty quantification, context influence, and semantic alignment to synchronously detect unfaithful sentences. By integrating efficiently measurable and complementary signals, SynCheck enables accurate and immediate feedback and intervention, achieving 0.85 AUROC in detecting faithfulness errors across six long-form retrieval-augmented generation tasks, improving prior best method by 4%. Leveraging SynCheck, we further introduce FOD, a faithfulness-oriented decoding algorithm guided by beam search for long-form retrieval-augmented generation. Empirical results demonstrate that FOD outperforms traditional strategies such as abstention, reranking, or contrastive decoding significantly in terms of faithfulness, achieving over 10% improvement across six datasets.

## 1 Introduction

Retrieval-augmented language models (RALMs) synergize large language models (LLMs) with external knowledge sources such as Wikipedia, web search, or tool invocations (Lewis et al., 2020; Guu et al., 2020; Borgeaud et al., 2022; Schick et al., 2023). Recent studies show that directly providing free-formed retrieved evidence in the context of LLMs can correct inaccurate or outdated parametric knowledge (Ram et al., 2023; Shi et al., 2024c), leading to strong performance on knowledge-intensive tasks such as open-domain question answering (Mallen et al., 2023) and long-form generation (Asai et al., 2024).

Despite the promising performance, the *trustworthiness* of RALMs' generation has become a concern: human evaluations reveal a substantial number of claims generated by RALMs contradicting with the provided context or cannot be grounded to any evidence (Niu et al., 2024; Wu et al., 2024). Such *unfaithful* use of knowledge by LLMs renders it difficult to trust the output even if the knowledge source is proven trustworthy. Towards a fully faithful and transparent use of knowledge, one line of work proposes *post-hoc* attribution or revision (Gao et al., 2023a,b), yet they are computationally expensive and could only be triggered after decoding. On the other hand, several *synchronous* decoding interventions have been proposed for critiquing and correcting RALMs' outputs on-the-fly, such as dynamic retrieval (Jiang et al., 2023b), reranking with fine-tuned critique tokens (Asai et al., 2024), and contrastive decoding for amplifying the influence of the knowledge (Shi et al., 2024a). However, these methods are mainly accuracy-oriented and it is unclear how well these signals are able to distinguish faithful samples from unfaithful ones. More importantly, these algorithms cannot provide any ways to guarantee or control the level of faithfulness of RALMs' final output.

To bridge these gaps, this paper undertakes a principled approach to faithfulness-oriented detection and decoding for long-form generation of RALMs. To start with, we compile a comprehensive benchmark to thoroughly evaluate faithfulness detectors at *sentence-level*, with the tasks covering biography generation, question answering, summarization, and data-to-text. Surprisingly, existing quality control methods in RALM systems including likelihood-based filtering (Jiang et al., 2023b) and instruction-tuned critique tokens (Asai et al., 2024) exhibit serious deficiency, only achieving approximately 0.6 AUROC across all the tasks (§5.1).



*Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 9390–9406
November 12-16, 2024 ©2024 Association for Computational Linguistics

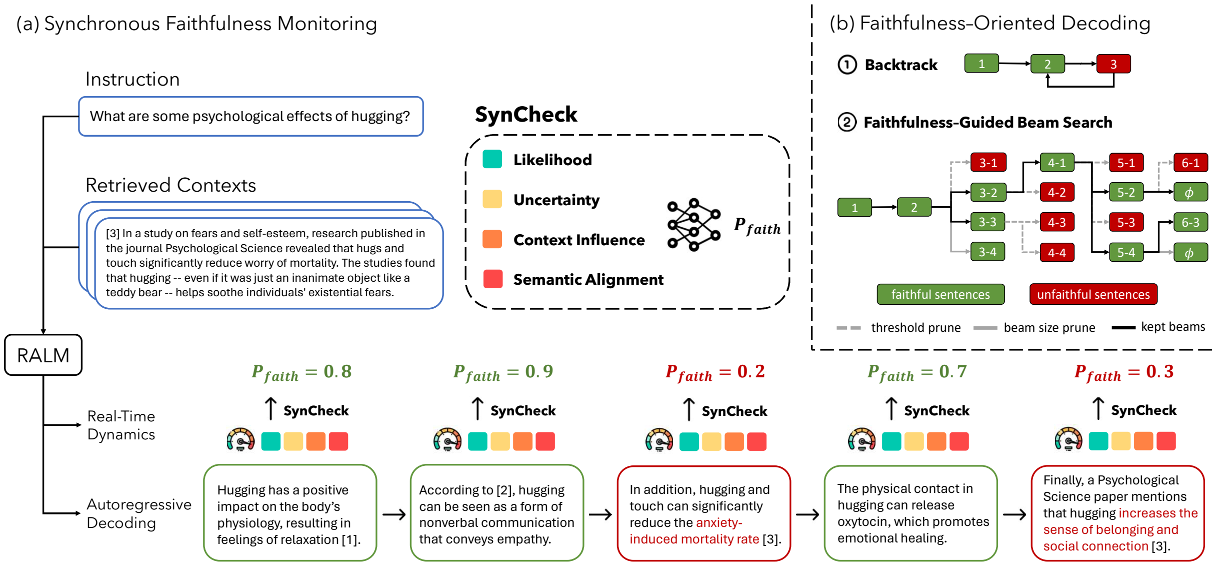


Figure 1: (a) An illustration of SYNCHECK, a fine-grained faithfulness checker for RALMs. SYNCHECK aggregates real-time decoding dynamics to accurately judge whether a sentence is trustworthy or not. (b) Leveraging SYNCHECK, we design a two-staged faithfulness-oriented decoding pipeline consisting of backtracking and beam search. In our algorithm, threshold prune takes place before beam size prune, and we use beam size 2 in the example.

In response, we propose SYNCHECK, a simple, efficient, yet accurate faithfulness error detector. To accurately characterize different types of faithfulness errors, SYNCHECK computes and monitors multiple complementary signals during RALM decoding: sequence likelihood, uncertainty quantification, context influence, and semantic alignment (Figure 1a). The signals are efficiently measured *synchronously* in the decoding process, and ensembled via a lightweight aggregator that imposes minimal overhead. Despite the efficient design, SYNCHECK achieves a strong faithfulness judgment performance, scoring on average over 0.85 AUROC across six datasets and two LLMs, outperforming five traditional baselines by ranging from 4% to 35% (§5.1). Furthermore, we find task-specific or model-specific training is not strictly required for training SYNCHECK's aggregator, which significantly improves its usability (§5.3).

With SYNCHECK, is it possible to design an intelligent and efficient algorithm to improve the generation's faithfulness with a guarantee? Traditional methods struggle to strike a balance between informativeness and faithfulness. For instance, abstention (Kamath et al., 2020; Ren et al., 2023; Feng et al., 2024) can be overly conservative, wasting the high quality part of the response. By contrast, contrastive decoding (Shi et al., 2024a) fails to enforce hard constraints to guarantee a basic level of faithfulness. We propose *faithfulness-oriented decoding* (FOD), a novel decoding algorithm that leverages SYNCHECK to *synchronously* monitor the generation faithfulness and guide the decoding process towards producing more faithful outputs. Specifically, FOD entails (1) backtracking at a low-quality sentence and (2) initiating a beam search that uses the faithfulness score to prune samples and guide the search direction (Figure 1b). Experiments show that FOD significantly improves the generation's faithfulness over greedy search (12%), abstention (10%), reranking (13%), and context-aware decoding (CAD, Shi et al. (2024a)) (19%) across six datasets. Compared to abstention, FOD improves in both faithfulness and informativeness. Compared to CAD, FOD generates more faithful samples at the same number of sentences (§5.2). We will publicly release the benchmark and our code at `https://github.com/xiaowu0162/sync-ralm-faithfulness`.

## 2 Related Work

**Context-Faithful LLMs** Characterizing and improving the context faithfulness of LLMs have been an important research topic. Longpre et al. (2021) identify over-reliance of language models on their parametric knowledge when presented with contexts contradicting with it. Xie et al. (2023) find that the persuasiveness of retrieved knowledge can be improved with an LM-centric knowledge verbalization process, yet LLMs still have inherent bias in choosing between conflicting pieces of knowledge. To improve the context faithfulness, common approaches include adapting the LLM to context-based generation (Shi et al., 2024b), improving the context quality (Xu et al., 2024), improving the decoding method (Zhou et al., 2023; Shi et al., 2024a), and post-hoc detection or revision (Niu et al., 2024; Gao et al., 2023a,b). Unfaithfulness to the context is also named as context-conflicting hallucinations (Zhang et al., 2023). Notable related works include detection with model features (Kadavath et al., 2022; Azaria and Mitchell, 2023; Yin et al., 2024; Jiang et al., 2024), decoding-stage interventions (Li et al., 2023; Chuang et al., 2024; Wang et al., 2024), and abstention (Kamath et al., 2020; Ren et al., 2023; Feng et al., 2024). In this work, we show that traditional approaches adopted by RALM systems cannot accurately distinguish faithful generated sentences from unfaithful ones. By utilizing both model-centric features and sematic alignment, SYNCHECK is able to characterize diverse types of unfaithfulness issues precisely, efficiently, and in a fine-grained manner.

**Advanced Decoding for RALMs** Recently, there has been a research interest in improving the decoding of black-box RALMs. Trivedi et al. (2023) and Shao et al. (2023) consider iterative retrieval for refining the generation. Shi et al. (2024a) use contrastive decoding to emphasize the influence of the retrieved context. FLARE (Jiang et al., 2023b) actively updates the context by setting a threshold on the likelihood of the least confident token in the generated sentence. Toolformer (Schick et al., 2023) learns to actively invoke tools at knowledge-intensive locations in generation. Self-RAG (Asai et al., 2024) learns critique tokens to score and rerank hypotheses generated based on different retrieved documents. Compared to prior work, our decoding algorithm utilizes SYNCHECK to give a fine-grained accurate guarantee of output quality, striking a better balance between faithfulness and informativeness of the output.

## 3 Approach

### 3.1 Problem Formulation

We consider retrieval augmented generation of free-form long responses. Let $\mathbf{x}$ be a sequence of input tokens encoding a question or an instruction. Let $\mathbf{c}$ denote a sequence of retrieved context tokens, which may include multiple free-form text chunks from multiple sources. Given the concatenation $[\mathbf{x}; \mathbf{c}]$, an LLM $\mathcal{M}$ predicts a sequence of segments $(\mathbf{s}_1, ..., \mathbf{s}_m)$ as the response, where each segment consists of a sequence of tokens[1].

Then, the task of *context faithfulness tracking* is defined as assigning a faithfulness label $\hat{y}_i \in \mathbb{R}$ to each newly generated segment $s_i$ based on $\mathbf{x}$, $\mathbf{c}$, and optionally the other segments. As many detection methods produce real-valued scores, we adopt AUROC as the evaluation metric.

### 3.2 SYNCHECK: Accurate Synchronous Faithfulness Monitoring for RALMs

Different from factuality, faithfulness imposes a unique focus on checking whether an RALM comprehends the retrieved contexts and avoids deviating from the contexts. Such behavior is thus challenging to characterize using simple heuristics employed by previous literature such as the generative likelihood (Jiang et al., 2023b). Instead, we introduce SYNCHECK, a fast and accurate feature-based detector for a range of untrustworthy generation scenarios on *an ascending hierarchy*: (1) the parametric and context knowledge is insufficient for forming the response, (2) the model fails to utilize the context in its predictions, and (3) the model uses the context unfaithfully. Concretely, SYNCHECK monitors four types of signals synchrnous to autoregressive decoding (Figure 1a):

**Likelihood** Low likelihood outputs often indicate the presence of *knowledge gaps*. When neither the parametric knowledge nor the retrieved information is sufficient, the model's response trivially bears faithfulness and trustworthiness issues. To detect this behavior, SYNCHECK measures the minimum likelihood as well as the length-normalized likelihood across all tokens in each sentence $s_i$.

**Uncertainty** A high predictive uncertainty suggests an *unconfident use of knowledge*. Whether the underlying cause is out-of-distribution questions, noisy retrieval, or a weak ability to incorporate the knowledge, the presence of high uncertainty strongly signals for a verification of the model's generation. SYNCHECK monitors the averaged token-level entropy within $s_i$ as well as the local intrinsic dimension of the activation of intermediate layers, which we hypothesize to more precisely characterize the degree of the LLM unfaithfully

[1] In this paper, we treat one sentence as a segment in our experiments, but our framework is applicable to any segment granularity (i.e., multi-sentence or sub-sentence).

mixing the retrieval context distribution with its parametric knowledge (Yin et al., 2024).

**Context Influence** An important behavior previous work failed to capture is the *over-dominance of parametric knowledge*. When an RALM heavily relies on its parametric knowledge, the generated information may deviate from the context and is thus largely non-attributable. Therefore, SYNCHECK monitors two token-wise distributions: $P_{\mathcal{M}}(s_i|\mathbf{x};\mathbf{c};s_{1:i-1})$ and $P_{\mathcal{M}}(s_i|\mathbf{x};s_{1:i-1})$, where $s_{1:i-1}$ is the sentences already generated by the RALM[2]. By contrasting the two distributions via token-level Kullback-Leibler divergence (Chang et al., 2023), we obtain informative indications of where $\mathbf{c}$ only has a weak influence on.

**Semantic Alignment** Even if the retrieved context exerts a high influence and the model produces a high confidence sentence, the output could still suffer from a *misinterpretation of context*. When the model makes such a mistake, it is hard to detect and correct with model-centric features, To complement with the previous features, SYNCHECK also runs a lightweight entailment checker (Zha et al., 2023) to gauge the likelihood of each $s_i$ being semantically inconsistent with the retrieved $\mathbf{c}$.

**SYNCHECK** For each $s_i$, we have collected a range of on-the-fly faithfulness signals. As they are designed to capture unique aspects, SYNCHECK trains a light-weight aggregator to learn the task-specific decision boundary with a small labelled dataset[3]. We explore three hypothesis spaces: logistic regression, XGBoost (Chen and Guestrin, 2016), and MLP. In the rest of the paper, they will be denoted as SYNCHECK$_{\text{LR}}$, SYNCHECK$_{\text{XGB}}$, and SYNCHECK$_{\text{MLP}}$ respectively.

In appendix A, we document the formulation and implementation details regarding the decoding-time feature collection and the aggregator training.

### 3.3 FOD: Faithfuless-Oriented Decoding

With the on-the-fly monitoring signals produced by SYNCHECK, can we further design effective intervention approaches to improve the faithfulness of RALMs' output? Abstention, or selective prediction, is a straightforward application: after detecting potential quality issues, the system can refuse

**Algorithm 1** **FOD: Faithfulness-Oriented Decoding**

**Require:** LLM, SYNCHECK, $\mathbf{x}$, $\mathbf{c}$, backtrack threshold $\tau_1$, sample pruning threshold $\tau_2$, sample size $S$, beam size $K$

// Stage 1: Greedy Search and Backtrack
out $\leftarrow$ [], $i \leftarrow 0$
// next sentence $s_i$, faithfulness score $f_i$
$s_i, f_i \leftarrow$ SYNCHECK ($\mathbf{x}$, $\mathbf{c}$, LLM($\mathbf{x}$; $\mathbf{c}$))
**while** $f_i \geq \tau_1$ **do**
  out $\leftarrow$ out + $[s_i]$, $i \leftarrow i+1$
  $s_i, f_i \leftarrow$ SYNCHECK ($\mathbf{x}$, $\mathbf{c}$, LLM($\mathbf{x}$; $\mathbf{c}$; $s_{0:(i-1)}$))
**end while**

// Stage 2: Faithfulness-Guided Beam Search
beams $\leftarrow$ [out], new_beams $\leftarrow$ []
**while** no beam has generated [EOS] **do**
  **for** $\mathcal{B}$ in beams **do**
    **for** $j = 1, ..., \lceil S/K \rceil$ **do**
      $s_i, f_i \leftarrow$ SYNCHECK ($\mathbf{x}$, $\mathbf{c}$, LLM($\mathbf{x}$; $\mathbf{c}$; $\mathcal{B}$))
      **if** $f_i \geq \tau_2$ **then**
        new_beams $\leftarrow \mathcal{B} + [s_i]$
      **end if**
    **end for**
  **end for**
  **if** new_beams **then**
    beams $\leftarrow$ $K$ most faithful beams in new_beams
    $i \leftarrow i+1$, new_beams $\leftarrow$ []
  **else**
    **break**
  **end if**
**end while**
**return** the most faithful beam in beams

to generate any output (Kamath et al., 2020; Ren et al., 2023; Feng et al., 2024). However, the coarse-grained binary abstention decision wastes considerable faithful information generated by the model. To devise a more principled way to enhance the faithfulness of the output while increasing the information retained, we introduce FOD, a *faithfulness-oriented decoding* algorithm for RALMs. Concretely, FOD integrates SYNCHECK to guide the search direction synchronously, with two stages:

1. Run greedy search until the first sentence $s_i$ with faithfulness score below a threshold $\tau_1$, which triggers the *backtrack* operation.
2. Starting from $s_{i-1}$, search in $K$ beams *in parallel*. In each step, sample several continuations from each of the beams, directly pruning out samples that have faithfulness score below a threshold $\tau_2$. Finally, retain $K$ beams with the highest aggregated faithfulness score.

Figure 1b presents an intuitive illustration of FOD, and the full algorithm is presented in Algorithm 1. We note that to calculate the faithfulness score of a partially decoded sequence, we always simply average across all the sentences in it. In addition, although we choose to halt the decoding when a single beam produces [EOS] as it gives empirically good performance, it is possible to con-

[2]Efficiency-wise, the second distribution indeed requires a separate forward pass. However, it could be calculated in parallel with the first distribution during token-level decoding.

[3]Empirically, the labelled set need not to be model-specific or task-specific, as we will explore in §5.3.

tinue the search until all the beams halt.

FOD's design optimizes the trade-off between faithfulness and informativeness. Compared to abstention, FOD is often able to retain a faithful prefix that already contains substantial information. Moreover, compared to contrastive decoding methods that operate on token-level logit (Shi et al., 2024a) which does not guarantees on the final output's quality, FOD provides a way to bound the minimum sentence-level faithfulness. Finally, the intermediate SYNCHECK signals also make the decoding process more human interpretable, facilitating further downstream explanation or debugging.

# 4 Experimental Setup

## 4.1 Task and Dataset Collection

We test SYNCHECK and FOD on a benchmark covering four common long-form RAG tasks: question answering (QA), summarization (Summ), data-to-text (Data2txt), and biography generation.

**RAGTruth** For QA, Summ, and Data2txt, we use the questions and the retrieved contexts provided by RAGTruth (Niu et al., 2024), which are respectively sourced from MS MARCO (Bajaj et al., 2016), CNN/Daily Mail (See et al., 2017), and Yelp Open Dataset (Yelp, 2021). Notably, questions in RAGTruth are designed for *long-form RAG*, which aligns well with our goal.

**Biography** We use the factscore benchmark (**FS**) introduced in Min et al. (2023). To simulate situations where unfaithful generations are more likely to occur, we construct two new datasets named famous-100 (**F-100**) and famous-100-anti (**F-100-anti**) where a model is asked to write a biography for 100 famous entities. The context for F-100 are retrievd from wikipedia, and for F-100-anti we create contexts from the evidence retrieved for another entity through entity substitution. By querying popular and salient parametric knowledge, F-100 and F-100-anti creates challenging scenarios for resisting to generate baseless/conflicting information.

**Split** We follow the train-test split in RAGTruth for QA, Summ, and Data2txt. FS, F-100, and F-100-anti only have a single test split. For SYNCHECK training, the respective train sets are used for QA, Summ, and Data2txt. The models for F-100 and F-100-anti are trained on FS, and the model for FS is trained on F-100. We further document the dataset construction, split details, as well as the basic statistics in appendix B.

## 4.2 Context Faithfulness Tracking

**Output Collection** We mainly test on Llama 2 7B Chat (Touvron et al., 2023) and Mistral 7B Instruct (Jiang et al., 2023a)[4]. For FS, F-100, and F-100-anti, we collect the outputs via greedy decoding. For QA, Summ, and Data2txt, we directly leverage the outputs provided by RAGTruth, which were sampled using various temperatures.

**Label construction** We use NLTK[5] to decompose the output into sentences, and assign each sentence with a faithfulness label. For QA, Summ, and Data2txt, we use the human-annotated baseless spans and conflict spans (Niu et al., 2024) as the unfaithful spans. For FS, F-100, and F-100-anti, we use a pre-trained propositionizer (Chen et al., 2023) to decompose the outputs into decontextualized propositions and then use an AutoAIS model (Honovich et al., 2022) to judge the faithfulness of each proposition. Finally, a lexical matching algorithm is used to map the span/proposition level faithfulness labels into sentence-level labels. We provide the detailed algorithm in appendix B.2.

**Baselines** We compare SYNCHECK with the following faithfulness checking baselines:

- SPANEXTRACT. Niu et al. (2024) proposes to instruct an evaluator LLM to directly predict the spans from the output corresponding to the unfaithful statements. We test GPT-4-Turbo and the fine-tuned Llama 2 13B model provided by Niu et al. (2024).
- CRITICTOK. Self-RAG (Asai et al., 2024) is an LLM instruction-tuned for critiquing RAG outputs. We leverage the model's critic token `[IsSup]` and report the score as the probability of `[fully supported]` divided by the sum of the probability of `[partially supported]` and `[no support]`, with each $s_i$ as the generated segment to critique.
- FLARE. Following Jiang et al. (2023b), we use the minimum likelihood across all the tokens in the generated sentence $s_i$.
- Lexical Alignment Models. We test ALIGNSCORE (Zha et al., 2023) and MINICHECK (Tang et al., 2024), both of which calculate a semantic alignment score between $\mathbf{C}$ and $s_i$.

[4]We use the model distributed at https://huggingface.co/mistralai/Mistral-7B-Instruct-v0.1.

[5]https://www.nltk.org/api/nltk.tokenize.PunktSentenceTokenizer.html

| Method | RAGTruth | | | Biography Generation | | | Average |
|---|---|---|---|---|---|---|---|
| | QA | Summ | Data2txt | FS | F-100 | F-100-anti | |
| **Llama 2 7B Chat** | | | | | | | |
| SPANEXTRACT$_{\text{GPT-4-Turbo}}$ ✣ | 0.705 | 0.773 | 0.794 | 0.630 | 0.548 | 0.506 | 0.659 |
| SPANEXTRACT$_{\text{Llama-2-13B}}$ ✣ | 0.734 | 0.688 | **0.842** | 0.552 | 0.660 | 0.657 | 0.689 |
| ALIGNSCORE ✣ | 0.772 | 0.768 | 0.679 | **0.791** | 0.897 | 0.886 | 0.799 |
| MINICHECK ✣ | 0.788 | 0.778 | 0.813 | 0.736 | 0.833 | 0.869 | 0.803 |
| CRITICTOK ✿ | 0.506 | 0.534 | 0.604 | 0.565 | 0.767 | 0.784 | 0.627 |
| FLARE ✿ | 0.631 | 0.527 | 0.532 | 0.649 | 0.694 | 0.677 | 0.618 |
| SYNCHECK$_{\text{LR}}$ ✿ | 0.812 | 0.753 | 0.775 | 0.771 | 0.908 | 0.893 | 0.819 |
| SYNCHECK$_{\text{XGB}}$ ✿ | 0.803 | 0.730 | 0.752 | 0.752 | 0.901 | 0.866 | 0.801 |
| SYNCHECK$_{\text{MLP}}$ ✿ | $\mathbf{0.833}_3$ | $\mathbf{0.787}_3$ | $0.785_2$ | $0.764_4$ | $\mathbf{0.918}_3$ | $\mathbf{0.896}_2$ | **0.831** |
| **Mistral 7B Instruct** | | | | | | | |
| SPANEXTRACT$_{\text{GPT-4-Turbo}}$ ✣ | 0.775 | 0.830 | 0.827 | 0.529 | 0.648 | 0.431 | 0.673 |
| SPANEXTRACT$_{\text{Llama-2-13B}}$ ✣ | 0.641 | 0.716 | **0.872** | 0.555 | 0.660 | 0.704 | 0.691 |
| ALIGNSCORE ✣ | 0.875 | 0.829 | 0.731 | **0.804** | 0.956 | 0.871 | 0.844 |
| MINICHECK ✣ | 0.824 | **0.874** | 0.832 | 0.761 | 0.852 | 0.880 | 0.837 |
| CRITICTOK ✿ | 0.588 | 0.578 | 0.529 | 0.564 | 0.870 | 0.838 | 0.661 |
| FLARE ✿ | 0.539 | 0.603 | 0.463 | 0.690 | 0.798 | 0.638 | 0.622 |
| SYNCHECK$_{\text{LR}}$ ✿ | **0.895** | 0.785 | 0.731 | 0.765 | **0.975** | 0.871 | 0.837 |
| SYNCHECK$_{\text{XGB}}$ ✿ | 0.877 | 0.750 | 0.819 | 0.769 | 0.865 | 0.867 | 0.825 |
| SYNCHECK$_{\text{MLP}}$ ✿ | $0.887_8$ | $0.829_0$ | $0.856_2$ | $0.773_2$ | $0.972_2$ | $\mathbf{0.883}_3$ | **0.867** |

Table 1: AUROC results of all context faithfulness tracking methods. We use superscripts to differentiate between external lexical detectors (✣) and methods using RALM-centric features (✿). We use subscript to denote the standard deviation across three runs. For instance, $0.887_8$ means a mean 0.887 and a standard deviation 0.008. SYNCHECK$_{\text{MLP}}$ achieves the strongest performance averaged across six tasks for both LLMs experimented.

In appendix C, we provide the implementation details of the baselines, including the prompt format for SPANEXTRACT and CRITICTOK.

### 4.3 Faithfulness Intervention

We use the same set of datasets and models to evaluate the proposed decoding method FOD.

**Evaluation Metric** We report two *response-level* metrics: faithfulness and informativeness. Inspired by Min et al. (2023), faithfulness is designed as proposition-level contextual consistency. The model proposed in Chen et al. (2023) is used to decompose the response into propositions, and we use `retrieval+llama+npm` method (Min et al., 2023) to factcheck each proposition directly with **C** as the context. Faithfulness is reported as the proportion of faithful propositions within all propositions. For informativeness, we report the number of propositions in the response. For the abstained or empty responses, we exclude them from faithfulness evaluation but assign 0 as the informativeness score.

**Baseline** We compare with (1) **abstention** - refusing to output when the faithfulness score predicted by SYNCHECK$_{\text{MLP}}$ for any of the output sentence below a certain threshold; (2) **reranking** - freely sampling the same number of responses as FOD would and directly return the sample with the best averaged faithfulness score from SYNCHECK; and (3) **CAD** (Shi et al., 2024a), a contrastive decoding method amplifying the influence from the context.

## 5 Results

### 5.1 Context Faithfulness Tracking

In Table 1, we report the AUROC of all the context faithfulness tracking approaches we consider. To begin with, we find that traditional methods adopted by RALMs systems, such as CRITICTOK and FLARE, only provide limited accuracy in identifying whether a single sentence is faithful or not. Moreover, SPANEXTRACT and ALIGNSCORE exhibit weak generalization performance across tasks. SPANEXTRACT performs poorly on Biography Generation tasks, which it has not been optimized on. ALIGNSCORE achieves decent performance on QA, Summ, and Biography as these tasks are similar to its extensive pre-training data (Zha et al., 2023). However, AlignScore fails to generalize to Data2txt. while requires fine-grained checking of details such as locations, ratings, and numeric information.

Among all the methods, SYNCHECK$_{\text{MLP}}$ achieves the strongest performance averaged across

| Method | QA | | Summ | | Data2txt | | FS | | F-100 | | F-100-anti | | Average | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | Faith. | Info. | Faith. | Info. | Faith. | Info. | Faith. | Info. | Faith. | Info. | Faith. | Info. | Faith. | Info. |
| **Llama 2 7B Chat** | | | | | | | | | | | | | | |
| Greedy | 0.628 | 10.9 | 0.716 | 9.4 | 0.440 | 13.3 | 0.489 | 8.0 | 0.766 | 9.5 | 0.603 | 7.6 | 0.607 | 9.8 |
| CAD | 0.710 | 10.4 | 0.755 | 9.5 | 0.468 | 14.0 | 0.278 | 7.4 | 0.611 | 22.4 | 0.476 | 19.2 | 0.549 | 13.8 |
| Abstention✿ | 0.639 | 3.5 | 0.712 | 9.0 | 0.422 | 8.2 | 0.509 | 2.8 | 0.809 | 2.3 | 0.584 | 1.1 | 0.613 | 3.7 |
| Reranking✿ | 0.745 | 9.3 | 0.757 | 9.8 | 0.474 | 13.4 | 0.466 | 10.4 | 0.621 | 10.0 | 0.563 | 12.9 | 0.604 | 11.0 |
| FOD (BT)✿ | 0.737 | 4.0 | 0.744 | 8.4 | **0.556** | 7.3 | **0.532** | 4.5 | 0.843 | 4.5 | 0.668 | 3.0 | 0.680 | 5.3 |
| FOD (Full)✿ | **0.768** | 4.3 | **0.770** | 9.5 | 0.529 | 10.4 | 0.507 | 5.1 | **0.870** | 4.8 | **0.735** | 3.5 | **0.697** | 6.3 |
| **Mistral 7B Instruct** | | | | | | | | | | | | | | |
| Greedy | 0.725 | 7.4 | 0.737 | 10.2 | 0.431 | 14.4 | 0.482 | 11.2 | 0.741 | 7.9 | 0.598 | 7.2 | 0.619 | 9.7 |
| CAD | 0.759 | 7.1 | **0.799** | 11.6 | 0.378 | 17.9 | 0.389 | 11.9 | 0.698 | 13.7 | 0.539 | 13.9 | 0.594 | 12.7 |
| Abstention✿ | 0.701 | 0.9 | 0.757 | 7.7 | 0.441 | 11.9 | 0.456 | 2.8 | 0.737 | 3.1 | 0.706 | 2.3 | 0.633 | 4.8 |
| Reranking✿ | 0.737 | 6.2 | 0.802 | 11.2 | 0.457 | 14.9 | 0.395 | 12.5 | 0.652 | 7.5 | 0.552 | 11.3 | 0.599 | 10.6 |
| FOD (BT)✿ | 0.781 | 1.3 | 0.790 | 6.3 | **0.603** | 4.7 | **0.510** | 8.2 | 0.756 | 6.2 | 0.692 | 4.5 | **0.688** | 5.2 |
| FOD (Full)✿ | **0.846** | 4.0 | 0.796 | 10.9 | 0.440 | 13.6 | 0.439 | 8.5 | **0.769** | 6.6 | **0.716** | 4.9 | 0.668 | 8.1 |

Table 2: Faithfulness-Informativeness evaluation results of faithfulness intervention methods. BT means backtracking only, without the following beam search. For all the results, we use a threshold 0.7 for abstention, and $\tau_1$=0.7, $\tau_2$=0.85 for our method, which we find generally work well. In addition, we use $K = 2$ and $S = 6$ for FOD (Full) and sample size 6 for reranking. We use ✿ to mark the decoding methods that leverage SYNCHECK.

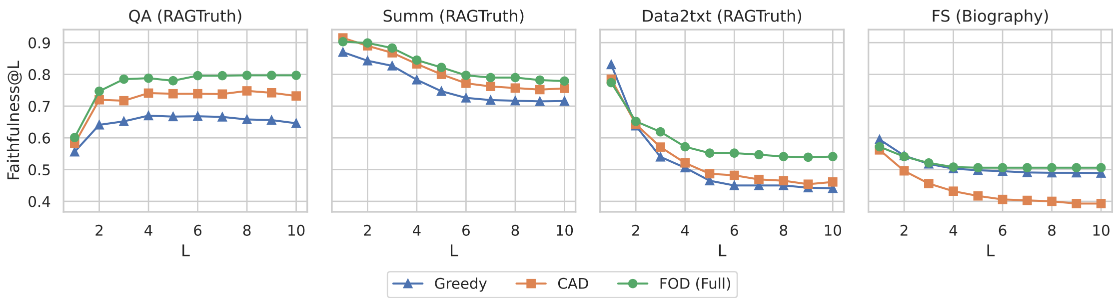


Figure 2: Faithfulness score on Llama 2 7B chat with output truncated to the first $L$ sentences.

all the tasks, with an average 0.831 AUROC for Llama 2 7B Chat and 0.867 for Mistral 7B Instruct. Notably, despite SYNCHECK leverages ALIGNSCORE for semantic alignment, it is able to outperform ALIGNSCORE on most of the tasks for both models. In §5.3, we provide further insights on feature importance as well as cross-task and cross-model generalization of SYNCHECK. Finally, among the hypothesis spaces we explored, we surprisingly find that simple logistic regression can achieve a very strong detection performance. This further highlights the value of the ensembled signals themselves, which enables high performance with simple and lightweight aggregators.

### 5.2 Intervention for Trustworthiness

Next, we study leveraging $\text{SYNCHECK}_{\text{MLP}}$ to improve the trustworthiness of the model output.

**Faithfulness vs. Informativeness** In Table 2, we evaluate the faithfulness-informativeness trade-off of different decoding strategies. Compared to the greedy search, FOD (BT) can already greatly improve the faithfulness. Compared to abstention, FOD (BT) improves *both* faithfulness and informativeness. For Summ, Data2txt, and FS, the response after backtracking still bears a number of propositions, indicated by the high informativeness. Through FOD (Full), the informativeness is further boosted without significantly impacting faithfulness. Notably, the hyperparameters for FOD ($\tau_1$, $\tau_2$, $K$, and $S$) are kept the same across all models and tasks, indicating its generalizability. Although CAD improves the faithfulness for QA, Summ, and Data2txt, it is ineffective for biography generation tasks. By comparison, FOD still consistently outperforms CAD for all of the six tasks and two models, except Summ for Mistral where the two algorithms have similar performance. Finally, compared to sampling and post-hoc reranking FOD significantly improve the faithfulness, indicating

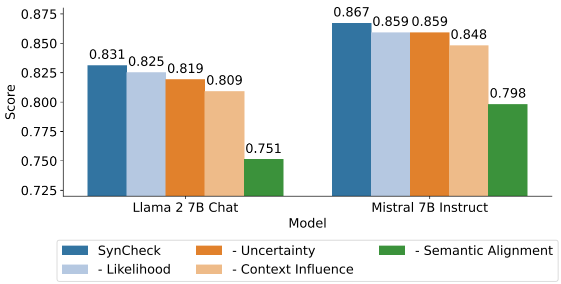


Figure 3: Feature ablation study with $\text{SYNCHECK}_{\text{MLP}}$ used as the detection model. We report the AUROC averaged across three runs with different random seeds.

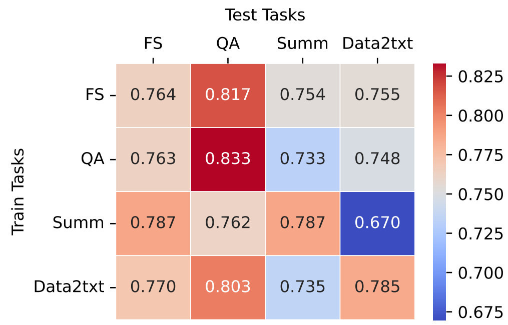


Figure 4: Performance of $\text{SYNCHECK}_{\text{MLP}}$ on different train-test task pairs using Llama 2 7B Chat. We report the AUROC averaged across three runs.

the necessity of performing on-the-fly monitoring and pruning of already unfaithful samples.

**Faithfulness@L** To check whether FOD truly provides a better faithfulness-informativeness trade-off, we further compare it with CAD in a more controlled setting. In Figure 2, we present the faithfulness of their predictions truncated to $L$ sentences. FOD consistently improves over greedy search and CAD across all the tasks, further confirming the effectiveness of the algorithm.

## 5.3 Analysis

In this section, we provide further analyses on SYNCHECK including feature ablation as well as cross-task/model faithfulness tracking.

**Ablation study** We conduct ablations on the features used by SYNCHECK, and present the results in Figure 3. Overall, we observe that removing each type of the proposed feature can harm the detection performance for both of the models studied. Although semantic alignment is the most important feature, it is also necessary to combine it with other dimensions to achieve the state-of-the-art performance. Context influence is the second important dimension, causing 0.02 to 0.03 absolute drop in AUROC when removed, suggesting its necessity and the unique value in discerning segments that are unfaithful to the context.

**Cross-Task Faithfulness Tracking** So far, we have assumed that task-specific data is required to train SYNCHECK. In Figure 4, we investigate the AUROC of $\text{SYNCHECK}_{\text{MLP}}$ trained on tasks other than the tested task. Overall, we find that detectors trained on one task can often transfer decently to another task. Specifically, most of the tasks can transfer well to detect unfaithful generations on QA. In addition, detectors trained on FS or Data2txt transfer well to most of the other tasks. Crucially, this result relaxes the dependence on task-specific data, leading to a more generalizable and data-efficient trustworthiness detection approach.

| | `Llama (13B)` | `Llama (70B)` | `Mistral (7B)` |
|---|---|---|---|
| QA | 0.854 | 0.853 | 0.870 |
| Summ | 0.811 | 0.739 | 0.821 |
| Data2txt | 0.718 | 0.706 | 0.779 |
| Bio | 0.782 | 0.759 | 0.777 |
| F-100 | 0.897 | 0.908 | 0.977 |
| F-100-anti | 0.866 | 0.865 | 0.906 |
| Average | 0.821 | 0.805 | 0.855 |

Table 3: Faithfulness tracking with $\text{SYNCHECK}_{\text{MLP}}$ and Llama 2 7B Chat as the surrogate model. `Llama` = `Llama 2 Chat`, `Mistral` = `Mistral Instruct`.

**Cross-Model Faithfulness Tracking** Finally, we explore using Llama 2 7B Chat as a *surrogate model* (Shrivastava et al., 2023) to perform faithfulness tracking for outputs generated by other models. As shown in Table 3, SYNCHECK trained on the surrogate model achieves a high performance on judging the faithfulness of other models' outputs. We hypothesize that although the samples may have a low likelihood to the surrogate model, other dimensions such as context influence and uncertainty still exhibit a clear boundary between faithful and unfaithful samples, making SYNCHECK applicable without access of the checked model or task-specific trustworthiness labels.

We further provide a comparison between FOD and single-feature reranking baselines in appendix D.1, a study for the hyperparameters $S$ and $K$ of FOD in appendix D.2, as well as several qualitative examples in appendix D.3.

# 6 Conclusion

In this paper, we introduced SYNCHECK, a real-time lightweight detector designed to enhance the faithfulness of RALMs in long-form generation by leveraging fine-grained decoding dynamics to detect unfaithful generated segments. Our empirical results demonstrate that SYNCHECK significantly outperforms existing faithfulness detection methods, achieving high AUROC scores across various tasks. Additionally, we proposed Faithfulness-Oriented Decoding, which improves the faithfulness and informativeness of RALM outputs compared to traditional faithfulness intervention methods. This work underscores the importance of real-time monitoring and targeted interventions in advancing RALMs as reliable tools for knowledge-intensive generation, paving the way for more trustworthy and interpretable RALM systems.

## Limitations

Despite the effectiveness of SYNCHECK and FOD, several limitations remain:

1. **Latency**. Despite the SYNCHECK operates logically concurrently with the decoding running it in beam search incurs extra latency. Specifically, in each step, $S$ sentences are sampled and have their faithfulness checked. With $S = 1$, the latency cost of FOD is exactly the same as CAD, since both of which incur an extra forward pass per token for calculating context influence. With $S$ GPUs, which is usually not a large number, FOD achieves the same latency as CAD.
2. **Segment Granularity**. In the paper, we mainly use sentences as the segment granularity, which is consistent with prior work such as (Jiang et al., 2023b). Future work could further confirm and improve the performance of FOD with segments of different granularity such as sub-sentence or multi-sentence.
3. **Diverse Real-World Tasks**. Finally, although we have performed evaluation on four representative tasks, further deploying the system to improve the quality of RAG on diverse real-world tasks in the wild is an exciting next step. In addition, it is also a promising investigation to employ SYNCHECK as a plug-and-play faithfulness monitor with RAG pipelines that feature larger models as the generation model.

## Ethics Statement

In this paper, we mainly investigate detecting and correcting the unfaithfulness in the outputs of RALMs. However, we are aware that faithfulness does not entail robustness, that a faithful RALM could be misled by biased resources and produce socially harmful claims. To build trustworthy RALM systems, we believe both the LLM and the retriever need the additional ability to critically judge the quality of the retrieved information before blindly following them. Finally, we are also aware that our approach improves the performance at the cost of additional computation. We are committed to further improving efficiency of our algorithm while maintaining the performance.

We access the officially released datasets for the RAGTruth datasets and the FS entities, both of which are released under the MIT license. We use these datasets as-is without any additional preprocessing process. As OpenAI models are involved in curating the F-100 and F-100-anti dataset, and a range of models are involved in creating the faithfulness tracking data, our code and datasets will be released with MIT license with a research-only use permission. In addition, we will not re-distribute the RAGTruth data but will instead redirect to the version distributed by the original authors.

## Acknowledgments

This research was supported in part by Taboola, CISCO, and NSF #2331966. We thank the Taboola team for the helpful discussion. We also thank I-Hung Hsu, Po-Nien Kung, Tanmay Parekh, as well as other members of the UCLA-NLP group and anonymous reviewers for their valuable feedback.

## References

Akari Asai, Zeqiu Wu, Yizhong Wang, Avirup Sil, and Hannaneh Hajishirzi. 2024. Self-RAG: Learning to retrieve, generate, and critique through self-reflection. In *The Twelfth International Conference on Learning Representations*.

Amos Azaria and Tom Mitchell. 2023. The internal state of an llm knows when its lying. *ArXiv preprint*, abs/2304.13734.

Payal Bajaj, Daniel Campos, Nick Craswell, Li Deng, Jianfeng Gao, Xiaodong Liu, Rangan Majumder, Andrew McNamara, Bhaskar Mitra, Tri Nguyen, et al. 2016. Ms marco: A human generated machine reading comprehension dataset. *ArXiv preprint*, abs/1611.09268.

Sebastian Borgeaud, Arthur Mensch, Jordan Hoffmann, Trevor Cai, Eliza Rutherford, Katie Millican, George van den Driessche, Jean-Baptiste Lespiau, Bogdan Damoc, Aidan Clark, Diego de Las Casas, Aurelia Guy, Jacob Menick, Roman Ring, Tom Hennigan, Saffron Huang, Loren Maggiore, Chris Jones, Albin Cassirer, Andy Brock, Michela Paganini, Geoffrey Irving, Oriol Vinyals, Simon Osindero, Karen Simonyan, Jack W. Rae, Erich Elsen, and Laurent Sifre. 2022. Improving language models by retrieving from trillions of tokens. In *International Conference on Machine Learning, ICML 2022, 17-23 July 2022, Baltimore, Maryland, USA*, volume 162 of *Proceedings of Machine Learning Research*, pages 2206–2240. PMLR.

Chung-Ching Chang, David Reitter, Renat Aksitov, and Yun-Hsuan Sung. 2023. Kl-divergence guided temperature sampling. *CoRR*, abs/2306.01286.

Tianqi Chen and Carlos Guestrin. 2016. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, August 13-17, 2016*, pages 785–794. ACM.

Tong Chen, Hongwei Wang, Sihao Chen, Wenhao Yu, Kaixin Ma, Xinran Zhao, Dong Yu, and Hongming Zhang. 2023. Dense x retrieval: What retrieval granularity should we use? *ArXiv preprint*, abs/2312.06648.

Yung-Sung Chuang, Yujia Xie, Hongyin Luo, Yoon Kim, James R Glass, and Pengcheng He. 2024. Dola: Decoding by contrasting layers improves factuality in large language models.

Shangbin Feng, Weijia Shi, Yike Wang, Wenxuan Ding, Vidhisha Balachandran, and Yulia Tsvetkov. 2024. Don't hallucinate, abstain: Identifying LLM knowledge gaps via multi-LLM collaboration. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 14664–14690, Bangkok, Thailand. Association for Computational Linguistics.

Luyu Gao, Zhuyun Dai, Panupong Pasupat, Anthony Chen, Arun Tejasvi Chaganty, Yicheng Fan, Vincent Zhao, Ni Lao, Hongrae Lee, Da-Cheng Juan, and Kelvin Guu. 2023a. RARR: Researching and revising what language models say, using language models. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 16477–16508, Toronto, Canada. Association for Computational Linguistics.

Tianyu Gao, Howard Yen, Jiatong Yu, and Danqi Chen. 2023b. Enabling large language models to generate text with citations. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 6465–6488, Singapore. Association for Computational Linguistics.

Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, and Ming-Wei Chang. 2020. Retrieval augmented language model pre-training. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, pages 3929–3938. PMLR.

Or Honovich, Roee Aharoni, Jonathan Herzig, Hagai Taitelbaum, Doron Kukliansy, Vered Cohen, Thomas Scialom, Idan Szpektor, Avinatan Hassidim, and Yossi Matias. 2022. TRUE: Re-evaluating factual consistency evaluation. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3905–3920, Seattle, United States. 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, et al. 2023a. Mistral 7b. *ArXiv preprint*, abs/2310.06825.

Che Jiang, Biqing Qi, Xiangyu Hong, Dayuan Fu, Yang Cheng, Fandong Meng, Mo Yu, Bowen Zhou, and Jie Zhou. 2024. On large language models' hallucination with regard to known facts. *ArXiv preprint*, abs/2403.20009.

Zhengbao Jiang, Frank Xu, Luyu Gao, Zhiqing Sun, Qian Liu, Jane Dwivedi-Yu, Yiming Yang, Jamie Callan, and Graham Neubig. 2023b. Active retrieval augmented generation. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 7969–7992, Singapore. Association for Computational Linguistics.

Saurav Kadavath, Tom Conerly, Amanda Askell, Tom Henighan, Dawn Drain, Ethan Perez, Nicholas Schiefer, Zac Hatfield-Dodds, Nova DasSarma, Eli Tran-Johnson, et al. 2022. Language models (mostly) know what they know. *ArXiv preprint*, abs/2207.05221.

Amita Kamath, Robin Jia, and Percy Liang. 2020. Selective question answering under domain shift. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5684–5696, Online. Association for Computational Linguistics.

Patrick S. H. 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 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*.

Kenneth Li, Oam Patel, Fernanda B. Viégas, Hanspeter Pfister, and Martin Wattenberg. 2023. Inference-time intervention: Eliciting truthful answers from a language model.

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.

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.

Sewon Min, Kalpesh Krishna, Xinxi Lyu, Mike Lewis, Wen-tau Yih, Pang Koh, Mohit Iyyer, Luke Zettlemoyer, and Hannaneh Hajishirzi. 2023. FActScore: Fine-grained atomic evaluation of factual precision in long form text generation. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 12076–12100, Singapore. Association for Computational Linguistics.

Cheng Niu, Yuanhao Wu, Juno Zhu, Siliang Xu, KaShun Shum, Randy Zhong, Juntong Song, and Tong Zhang. 2024. RAGTruth: A hallucination corpus for developing trustworthy retrieval-augmented language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 10862–10878, Bangkok, Thailand. Association for Computational Linguistics.

Ori Ram, Yoav Levine, Itay Dalmedigos, Dor Muhlgay, Amnon Shashua, Kevin Leyton-Brown, and Yoav Shoham. 2023. In-context retrieval-augmented language models. *Transactions of the Association for Computational Linguistics*, 11:1316–1331.

Jie Ren, Jiaming Luo, Yao Zhao, Kundan Krishna, Mohammad Saleh, Balaji Lakshminarayanan, and Peter J. Liu. 2023. Out-of-distribution detection and selective generation for conditional language models. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net.

S. E. Robertson and S. Walker. 1994. Some simple effective approximations to the 2-poisson model for probabilistic weighted retrieval. In *SIGIR '94*, pages 232–241, London. Springer London.

Timo Schick, Jane Dwivedi-Yu, Roberto Dessì, Roberta Raileanu, Maria Lomeli, Eric Hambro, Luke Zettlemoyer, Nicola Cancedda, and Thomas Scialom. 2023. Toolformer: Language models can teach themselves to use tools.

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.

Zhihong Shao, Yeyun Gong, Yelong Shen, Minlie Huang, Nan Duan, and Weizhu Chen. 2023. Enhancing retrieval-augmented large language models with iterative retrieval-generation synergy. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 9248–9274, Singapore. Association for Computational Linguistics.

Weijia Shi, Xiaochuang Han, Mike Lewis, Yulia Tsvetkov, Luke Zettlemoyer, and Wen-tau Yih. 2024a. 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.

Weijia Shi, Sewon Min, Maria Lomeli, Chunting Zhou, Margaret Li, Xi Victoria Lin, Noah A. Smith, Luke Zettlemoyer, Wen-tau Yih, and Mike Lewis. 2024b. In-context pretraining: Language modeling beyond document boundaries. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net.

Weijia Shi, Sewon Min, Michihiro Yasunaga, Minjoon Seo, Richard James, Mike Lewis, Luke Zettlemoyer, and Wen-tau Yih. 2024c. REPLUG: Retrieval-augmented black-box language models. 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 8371–8384, Mexico City, Mexico. Association for Computational Linguistics.

Vaishnavi Shrivastava, Percy Liang, and Ananya Kumar. 2023. Llamas know what gpts don't show: Surrogate models for confidence estimation. *ArXiv preprint*, abs/2311.08877.

Liyan Tang, Philippe Laban, and Greg Durrett. 2024. Minicheck: Efficient fact-checking of llms on grounding documents. *arXiv preprint arXiv:2404.10774*.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *ArXiv preprint*, abs/2307.09288.

Harsh Trivedi, Niranjan Balasubramanian, Tushar Khot, and Ashish Sabharwal. 2023. Interleaving retrieval with chain-of-thought reasoning for knowledge-intensive multi-step questions. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 10014–10037, Toronto, Canada. Association for Computational Linguistics.

Ante Wang, Linfeng Song, Baolin Peng, Ye Tian, Lifeng Jin, Haitao Mi, Jinsong Su, and Dong Yu. 2024. Fine-grained self-endorsement improves factuality and reasoning. *ArXiv preprint*, abs/2402.15631.

Kevin Wu, Eric Wu, Ally Cassasola, Angela Zhang, Kevin Wei, Teresa Nguyen, Sith Riantawan, Patricia Shi Riantawan, Daniel E Ho, and James Zou. 2024. How well do llms cite relevant medical references? an evaluation framework and analyses. *ArXiv preprint*, abs/2402.02008.

Jian Xie, Kai Zhang, Jiangjie Chen, Renze Lou, and Yu Su. 2023. Adaptive chameleon or stubborn sloth: Revealing the behavior of large language models in knowledge conflicts. In *The Twelfth International Conference on Learning Representations*.

Fangyuan Xu, Weijia Shi, and Eunsol Choi. 2024. RECOMP: improving retrieval-augmented lms with context compression and selective augmentation. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net.

Yelp. 2021. Yelp open dataset.

Fan Yin, Jayanth Srinivasa, and Kai-Wei Chang. 2024. Characterizing truthfulness in large language model generations with local intrinsic dimension. In *Forty-first International Conference on Machine Learning, ICML 2024, Vienna, Austria, July 21-27, 2024*. OpenReview.net.

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.

Yue Zhang, Yafu Li, Leyang Cui, Deng Cai, Lemao Liu, Tingchen Fu, Xinting Huang, Enbo Zhao, Yu Zhang, Yulong Chen, et al. 2023. Siren's song in the ai ocean: a survey on hallucination in large language models. *ArXiv preprint*, abs/2309.01219.

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.

# Supplementary Material: Appendices

## A SynCheck Details

### A.1 Decoding-time feature collection

| Feature Type | Alias | Range |
|---|---|---|
| Likelihood | min_prob | $[0, 1]$ |
| | mean_prob | $[0, 1]$ |
| Uncertainty | max_entropy | $[0, 1]$ |
| | mean_entropy | $[0, 1]$ |
| | lid_layer_15 | $[0, +\infty)$ |
| | lid_layer_16 | $[0, +\infty)$ |
| | lid_layer_17 | $[0, +\infty)$ |
| Context Influence | mean_contrastive_kl | $[0, 1]$ |
| | large_kl_pos | $[0, +\infty)$ |
| Semantic Alignment | align_score | $[0, 1]$ |

Table 4: A list of features monitored by SynCheck.

In Table 4, we outline the full list of features monitored and leveraged by SynCheck. In the rest of this section, we detail the methodology used to measure each feature.

**min_prob and mean_prob** For each of the predicted token in $s_i$, we calculate its likelihood. `min_prob` takes the likelihood of the most unlikely token and `mean_prob` takes the average.

**mean_entropy and max_entropy** For each sentence $s_i$, we calculate the entropy of the distribution over the entire vocabulary for each of the output token position. `mean_entropy` takes the average of these values and `max_entropy` takes the max.

**lid_layer_x** When the model finishes decoding for a single segment $S = [s_0, s_1, ..., s_n]$, we collect the hidden representation of the last token $s_n$ from layer x. Then, we calculate the local intrinsic dimension using the representation against a set of T pre-computed hidden dimensions measured in the same way using labelled segments from the train set, following the LID-MLE method proposed in Yin et al. (2024). Specifically, we sort the set of T reference points by their distance to $s_n$ and calculate local intrinsic dimension as $LID(S) = \frac{1}{T-1}(\sum_{j=1}^{T} \log(D(s_n, q_T)/D(s_n, q_j)))^{-1}$, where $D(s_n, q_j)$ is the Euclidean distance from the $s_n$ representation to the $j$-th reference point. Untruthful sentences will have larger LID values. In preliminary studies, we found `lid_layer_x` to be ineffective for Summ and Data2txt. Therefore, we only incorporate this set of features for FS, F-100, F-100-anti, and QA.

**mean_contrastive_kl and large_kl_pos** As the model decodes each token in the sentence $S = [s_0, s_1, ..., s_n]$, we simultaneously collect the distribution over the vocabulary as well as the distribution with the same generation prefix but without the retrieved context. Concretely, given the LLM $\mathcal{M}$, a query $X$, a context $C$, a output prefix $O$, then the two distributions could be written as $D_{i1} = P_{\mathcal{M}}(s_i|X; C; O; [s_0, \ldots, s_{i-1}])$ and $D_{i2} = P_{\mathcal{M}}(s_i|X; O; [s_0, \ldots, s_{i-1}])$. Then, we obtain a sequence of KL-divergence values $KL(D_{i1}||D_{i2})$ for S. The feature `mean_contrastive_kl` the average across all positions in this sentence. The feature `mean_contrastive_kl` the average across all positions in this sentence. `large_kl_pos` is a discrete version of the mean signal which counts the number of positions with the KL-divergence greater than 3.0.

**align_score** For each sentence, we calculate its AlignScore (Zha et al., 2023) with the context as the reference. We use the AlignScore-base model released by the authors for its efficiency.

For all the feature collection, classification, and decoding experiments, we execute them on a local server with 8 Nvidia A6000 GPUs. For models of size 7B, the latency of running the full decoding pipeline is roughly 30s per sample using two GPUs. To run the end-to-end decoding and evaluation on the test set, the latency is roughly 2 hours per task.

### A.2 Aggregator training details

We use the `scikit-learn` and the `xgboost` Python package to train different variations of SynCheck. For $\text{SynCheck}_{\text{MLP}}$, we use a two-layer neural network with hidden dimension 100 and ReLU activation. We use a learning rate of 0.001, Adam optimizer, batch size 128, and train for 300 iterations. While we did not extensively tune the hyperparameters, we did a preliminary study with learning rate {0.01, 0.001, 0.0001} and batch size {32, 64, 128} which supports the chosen set of hyperparameters.

## B Benchmark Construction Details

### B.1 F-100 and F-100-anti

To create F-100 and F-100-anti, we first directly prompt `ChatGPT` to propose 100 entities representing famous people. Figure 5 presents a full list of the entities. Then, we run BM25 retrieval (Robertson and Walker, 1994) on paragraph-chunked Wikipedia data (version 2021-10-13, processed by Chen et al. (2023)) with "Tell me a bio of [entity]" as the query. For F-100, we keep and use the top-10 paragraphs as the retrieved evidence. For F-100-anti, we link each entity E1 with the contexts retrieved for another entity E2, while replacing all the occurrences of E2 with E1.

Nikita Khrushchev, Malcolm X, John F. Kennedy, Bill Gates, Queen Elizabeth II, Napoleon Bonaparte, Mohandas Gandhi, Albert Einstein, Cleopatra, Adolf Hitler, Freddie Mercury, Plato, J.K. Rowling, Karl Marx, Margaret Thatcher, Angela Merkel, Vincent Van Gogh, John Lennon, Che Guevara, Cristiano Ronaldo, Rosa Parks, Rihanna, Alexander the Great, Isaac Newton, Julius Caesar, Amelia Earhart, Simone Biles, Michael Jordan, Elton John, Sigmund Freud, Joseph Stalin, Jane Goodall, Beyoncé, Adele, Charles Dickens, Thomas Edison, Eminem, Virginia Woolf, Taylor Swift, Tupac Shakur, Justin Bieber, Tim Berners-Lee, Kanye West, Marie Curie, Ludwig van Beethoven, Pablo Neruda, Steve Jobs, Usain Bolt, Bruce Lee, Marilyn Monroe, Michael Jackson, Shakira, Mozart, Lady Gaga, Vladimir Putin, Charles Darwin, Harriet Tubman, Benjamin Franklin, Oprah Winfrey, Malala Yousafzai, Socrates, Mahatma Gandhi, Pablo Picasso, Frida Kahlo, Nelson Mandela, Whitney Houston, Winston Churchill, Stephen Hawking, Fidel Castro, Margaret Atwood, Madonna, Leonardo DiCaprio, Elizabeth I, Galileo Galilei, Muhammad Ali, Mao Zedong, William Shakespeare, Joan of Arc, George Washington, Mikhail Gorbachev, Abraham Lincoln, Pele, Martin Luther King Jr., Jane Austen, Ed Sheeran, Sachin Tendulkar, Ariana Grande, Wolfgang Amadeus Mozart, Mark Zuckerberg, Vincent van Gogh, Bob Marley, Ronald Reagan, Barack Obama, Britney Spears, Walt Disney, Leonardo da Vinci, Elvis Presley, Lionel Messi, Anne Frank, Confucius

Figure 5: A list of the well-known entities included in the F-100 and F-100-anti biography generation dataset.

### B.2 Trustworthiness label generation

In this section, we describe the lexical matching procedure we design for obtaining sentence-level faithfulness label.

**RAGTruth** For QA, Summ, and Data2txt, the hallucination spans annotated by RAGTruth are in the format (start, end) indication the character-level boundary. For each sentence in the output, the sentence is then labelled as unfaithful if either (1) it contains one or more hallucination spans or (2) it is contained by a hallucination span.

**Biography** For FS, F-100, and F-100-anti, the outputs are converted to propositions and judged by the AutoAIS system[6]. We map the propositions back to sentence-level labels using token recall. Specifically, for each unfaithful proposition, we find the sentence that contains the highest proportion of its tokens and label it as unfaithful. The rest sentences are labelled as faithful. We conduct a small-scale human inspection and find that this approach generally produces correct label mappings.

### B.3 Dataset Statistics

We further outline the dataset statistics in Table 5, including the instance count, retrieval length, generation length, as well as the label distribution.

## C Implementation Details of Baselines

In this section, we further describe the implementation details of two context faithfulness tarcking baselines we compare with SYNCHECK.

**SPANEXTRACT** For QA, Summ, and Data2txt, we follow the evaluation prompt in the original paper (Niu et al., 2024). For the biography generation tasks, we use the following evaluation prompt, which share some similarity with the original prompt for evaluating QA.

```
Passages: {Retrieved Context}
Question: Tell me a bio of {entity}.
Answer: {answer}
Please analyze the provided Passages, Question, and Answer, and indicate whether there are any hallucinated (invented or incorrect) parts in the answer. If there are, please specify the exact span that is hallucinated. Please provide a response in the following format:...(json format) ...
```

**CRITICTOK** For Self-RAG, we refer to the original implementation and use the following prefix for scoring each segment:

```
{Instruction} [Retrieve] {Retrieved Context} [Relevant] {Response Segment}
```

We also tried using the full response until the segment instead of only a single segment. However, we find that the previous approach performs slightly better and thus select it as the baseline.

```
{Instruction} [Retrieve] {Retrieved Context} [Relevant] {Full Response until Segment}
```

[6]We use the model released at https://huggingface.co/google/t5_xxl_true_nli_mixture.

| Dataset | Split | # Docs | # Instances | # Cxt Sents | # Gen Sents | % Faithful Sents | % Faithful Instances |
|---|---|---|---|---|---|---|---|
| | | | **Llama 2 7B Chat** | | | | |
| **QA** | train | 839 | 9669 | 17.7 | 14.98 | 87.06% | 54.47% |
| | test | 150 | 1453 | 17.1 | 13.34 | 90.71% | 34.67% |
| **Summ** | train | 793 | 3771 | 35.9 | 5.43 | 88.17% | 48.30% |
| | test | 150 | 692 | 34.8 | 5.05 | 91.47% | 33.33% |
| **Data2txt** | train | 883 | 5750 | 23.8 | 7.13 | 77.51% | 86.64% |
| | test | 150 | 940 | 24.3 | 6.65 | 80.53% | 82.00% |
| **FS** | test | 500 | 1770 | 51.5 | 4.46 | 73.62% | 51.40% |
| **F-100** | test | 100 | 405 | 38.2 | 4.28 | 56.79% | 67.00% |
| **F-100-anti** | test | 100 | 349 | 36.6 | 4.01 | 53.58% | 73.00% |
| | | | **Mistral 7B Instruct** | | | | |
| **QA** | train | 839 | 6055 | 18.6 | 11.99 | 88.16% | 41.24% |
| | test | 150 | 882 | 18.1 | 10.60 | 91.84% | 20.67% |
| **Summ** | train | 793 | 4530 | 33.9 | 6.40 | 84.55% | 66.83% |
| | test | 150 | 838 | 33.0 | 6.13 | 86.52% | 57.33% |
| **Data2txt** | train | 883 | 7677 | 23.0 | 9.38 | 79.77% | 93.32% |
| | test | 150 | 1269 | 23.5 | 9.13 | 82.74% | 88.67% |
| **FS** | test | 290 | 1629 | 51.9 | 6.37 | 83.00% | 58.28% |
| **F-100** | test | 61 | 227 | 38.0 | 3.94 | 79.30% | 40.98% |
| **F-100-anti** | test | 62 | 214 | 34.8 | 3.76 | 55.14% | 62.90% |

Table 5: Basic statistics of our evaluation benchmark. Notably, our benchmark tests long-form generation with long-form retrieved evidence. # Cxt Sents and # Gen Sents stand for the number of sentences in the context and the model's output. Both the tested models exhibit a considerable rate of unfaithful sentences.

# D Further Analysis

## D.1 Single-feature reranking baselines

To further verify the effectiveness of the proposed detection and decoding algorithms, we extend table 2 with two additional baselines: reranking with either only semantic alignment or only likelihood. These baselines are directly comparable with the line "Reranking✿" in table 2, which is copied to the third line of table 6. As we can observe, reranking with SYNCHECK outperforms reranking with these individual features. The full FOD algorithm further improves the output's faithfulness.

## D.2 Hyperparameter for FOD

In this section, we provide further studies on the hyperparameter sensitivity for FOD. Specifically, we use the Llama 2 7B Chat model and investigate two hyperparameters: the beam size $B$ and the sample size $S$. We keep the other parameters the same as unsed in the main experiments.

We present the results in Figure 6 and Figure 7. For the beam size, we observe that a higher beam size generally decreases the informativeness. This could be due to more easily achieving the end-of-sequence token with more diverse exploration patterns. Meanwhile, the faithfulness change under beam size changes varies task-by-task. For the sample size, we find a positive correlation with the informativeness and an U-shaped behavior for faithfulness. Generally, combination of modest sample size and modest beam size works well.

## D.3 Qualitative Study

In Figure 8, we present examples collected from QA, Summ, Data2txt, and FS using Llama 2 7B Chat. SynCheck provides an accurate detection of all the unfaithful sentences. In addition, it is able to recognize the faithful sentences following unfaithful ones, enabling the lookahead and the selection interventions after backtracking.

| Method | QA | | Summ | | Data2txt | | F-100 | | Average | |
|---|---|---|---|---|---|---|---|---|---|---|
| | Faith. | Info. | Faith. | Info. | Faith. | Info. | Faith. | Info. | Faith. | Info. |
| `Llama 2 7B Chat` | | | | | | | | | | |
| Reranking (Likelihood) | 0.686 | 8.5 | 0.731 | 9.5 | 0.468 | 14.1 | 0.513 | 11.9 | 0.600 | 11.0 |
| Reranking (AlignScore) | 0.738 | 8.8 | 0.731 | 9.3 | 0.468 | 13.3 | 0.538 | 9.7 | 0.619 | 10.3 |
| Reranking✿ | 0.745 | 9.3 | 0.757 | 9.8 | 0.474 | 13.4 | 0.563 | 10.0 | 0.635 | 0.106 |
| FOD (Full)✿ | **0.768** | 4.3 | **0.770** | 9.5 | **0.529** | 10.4 | **0.735** | 0.48 | **0.701** | 7.3 |

Table 6: Further Faithfulness-Informativeness evaluation results of faithfulness intervention methods. We further compare the best results in table 2 with two baselines of reranking with a single feature. We use ✿ to mark the decoding methods that leverage SYNCHECK.

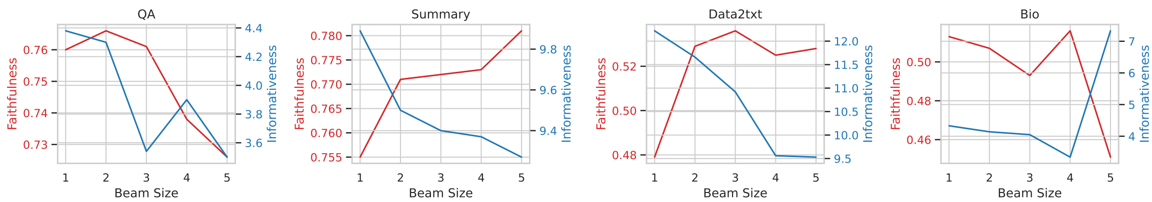


Figure 6: Faithfulness and Informativeness of Llama 2 7B Chat as a function of the beam size.

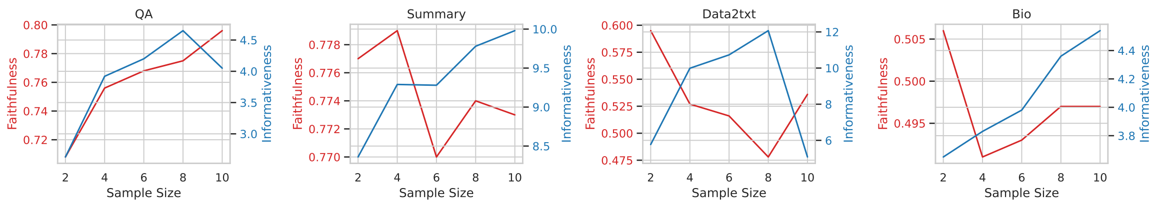


Figure 7: Faithfulness and Informativeness of Llama 2 7B Chat as a function of the sample size.

**Instruction:** Republicans use ________ to allocate delegates in their presidential nominating procedure quizlet
**Context:** [1] Some states award delegates to candidates on a winner-take-all basis, meaning that the candidate with the most votes in a state is awarded all of that state's delegates. Other states award delegates in proportion to each candidate's share of the primary vote. [2] To win the nomination, a candidate must win the votes of at least 1,191 delegates at the convention. Sen. John McCain (R-AZ) reached that benchmark in March 2008, becoming the Republican Party's presumptive nominee. [3] Most will have already pledged to whichever candidate won their state. To win the nomination, a candidate must win the votes of at least 1,191 delegates at the convention. Sen. John McCain (R-AZ) reached that benchmark in March 2008, becoming the Republican Party's presumptive nominee.
**Response:** Based on the provided passages, Republicans use a proportional method to allocate delegates in their presidential nominating procedure. [SynCheck: 0.849] According to passage 1, some states award delegates to candidates on a winner-take-all basis, while others award delegates in proportion to each candidate's share of the primary vote. [SynCheck: 0.964] This suggests that the allocation of delegates is not solely determined by a simple majority vote, but rather takes into account the distribution of support among the various candidates. [SynCheck: 0.302] Therefore, the answer to the question is: Proportional. [SynCheck: 0.827]

**Instruction:** Summarize the following news within 53 words
**Context:** Their relationship led to jail time for her, but Mary Kay Letourneau Fualaau wants the world to know that she and her husband, Vili Fualaau, are still happily together. She was a married 34-year-old teacher and mother of four in Seattle in 1996 when she began an affair with Fualaau, her 13-year-old student. Letourneau gave birth to her young lover's child and went on to serve more than seven years in prison on charges related to their sexual relationship. On Friday, ABC's "20/20" will air an exclusive interview with the couple, who will celebrate their 10th wedding anniversary on May 20. The pair wed soon after she was released from prison in 2005 and are now the parents of two teen girls. According to ABC, during the interview with Barbara Walters, "Mary Kay tells Walters what makes their marriage work in spite of their huge difference in their age." "She also discusses her surprising plans to teach again, as well as her intentions to have her status as a registered sex offender lifted," a posting on the network's site says. "Vili Fualaau, meanwhile, discusses his bouts with alcoholism, depression and why he believes the system failed him while he was still a minor." Letourneau Fualaau is now 53, and her husband is 31.
**Response:** Mary Kay Letourneau, who served 7 years in prison for having an affair with her 13-year-old student, is still married to her husband Vili Fualaau and they are celebrating their 10th wedding anniversary this May. [SynCheck: 0.974] Despite their large age gap, they claim their marriage works due to their shared values and love for each other. [SynCheck: 0.627] Letourneau hopes to become a teacher again and has applied to have her sex offender status lifted. [SynCheck: 0.945]

**Instruction:** Write an objective overview about the following local business based only on the provided structured data.
**Context:** {'name': 'Super Cucas', 'address': '6527 Madrid Rd', 'city': 'Isla Vista', 'state': 'CA', 'categories': 'Ethnic Food, Specialty Food, Local Flavor, Mexican, Restaurants, Food, Breakfast & Brunch, Fast Food, Juice Bars & Smoothies', 'hours': {'Monday': '7:0-0:0', 'Tuesday': '7:0-0:0', 'Wednesday': '7:0-0:0', 'Thursday': '7:0-0:0', 'Friday': '7:0-0:0', 'Saturday': '7:0-0:0', 'Sunday': '7:0-0:0'}, 'attributes': {'BusinessParking': {'garage': False, 'street': True, 'validated': False, 'lot': None, 'valet': False}, 'RestaurantsReservations': False, 'OutdoorSeating': None, 'WiFi': 'no', 'RestaurantsTakeOut': True, 'RestaurantsGoodForGroups': True, 'Music': None, 'Ambience': {'touristy': False, 'hipster': False, 'romantic': False, 'divey': None, 'intimate': False, 'trendy': False, 'upscale': False, 'classy': False, 'casual': True}}, 'business_stars': 3.5, 'review_info': [{'review_stars': 4.0, 'review_date': '2021-11-03 21:59:35', 'review_text': "Super Cucas is really tasty and the portion sizes are so generous. They're also open late which is always great when you're in a college town. I love that they have a menu that offers every style of meat veganized (including soy pastor!) I'd love to try the vegan fries with meat (asada fries) and the breakfast burritos on my next visit. The portions are huge so I would recommend splitting or saving for 2 meals."}]}
**Response:** Super Cucas is a locally-owned restaurant located in Isla Vista, California, serving a variety of Ethnic, Specialty, and Mexican cuisine. [SynCheck: 0.939] With a rating of 3.5 stars, the restaurant is popular among locals and visitors alike, particularly for its late hours of operation (until 10 PM from Monday to Saturday). [SynCheck: 0.594] The establishment is known for its generous portion sizes and vegetarian options, including veganized meats. [SynCheck: 0.988] While some reviewers have praised the quality of the food, others have experienced poor customer service, with instances of incorrect orders and rude staff behavior. [SynCheck: 0.801] Nonetheless, Super Cucas remains a popular spot for those seeking tasty and affordable meals in the area. [SynCheck: 0.794]

**Instruction:** Who is Blanca Errázuriz?
**Context:** [1] Blanca Errázuriz Blanca Errázuriz Blanca Elena Errázuriz Vergara (9 April 1894 - 20 March 1940), also known as Bianca de Saulles, was a Chilean socialite and the former wife of football player and businessman John de Saulles. In August 1917, Errázuriz fatally shot de Saulles multiple times during a disagreement over the custody of their son. After a highly publicized and sensational trial, Errázuriz was acquitted of her ex-husband's murder. Errázuriz was born in Via del Mar, Chile, the eldest daughter of Guillermo Errázuriz Urmeneta and of Blanca Vergara Alvarez, a beauty known as the Star of Santiago. [2] Blanca Errázuriz press, and the champion of the suffragettes who portrayed her as the victim of the chauvinism prevalent in the society of the time. Blanca Errázuriz was unanimously acquitted of the murder charges on 1 December 1917 in what was called a "popular" verdict. Following the trial, Errázuriz moved to San Francisco where she sought and obtained full custody of her son. She and her son went later to Japan. Eventually she and her son returned to Chile and settled there. On 22 December 1921, she remarried, this time to engineer Fernando Santa Cruz Wilson in Santiago; the couple later divorced. ... (8 documents omitted)
**Response:** Blanca Errázuriz was a Chilean socialite and the former wife of football player and businessman John de Saulles. [SynCheck: 0.954] She was born in Viña del Mar, Chile in 1894 and was known for her beauty and her marriage to de Saulles, who was 15 years her senior. [SynCheck: 0.033] In 1917, Errázuriz fatally shot de Saulles during a disagreement over the custody of their son, and she was acquitted of his murder after a highly publicized and sensational trial. [SynCheck: 0.912] Errázuriz was also known for her friendship with Rudolph Valentino and her association with the politically influential Errázuriz family, of Basque descent. [SynCheck: 0.654]

Figure 8: Examples from Llama 2 7B Chat on QA, Summ, Data2txt, and FS. We color faithful sentences in blue and unfaithful ones in red. SYNCHECK provides accurate detection of the locations of unfaithful output.