Atomic Consistency Preference Optimization for Long-Form Question Answering

Jingfeng Chen* jingfenc@andrew.cmu.edu

Raghuveer Thirukovalluru* raghuveer.thirukovalluru@duke.edu

Junlin Wang junlin.wang2@duke.edu

Luo Kaiwei luokw1@chinatelecom.cn

Bhuwan Dhingra bhuwan.dhingra@duke.edu

Abstract

Large Language Models (LLMs) often produce factoid hallucinations - plausible yet incorrect answers. A common mitigation strategy is model alignment, which improves factual accuracy by training on curated (factual, non-factual) pairs. However, this approach often relies on a stronger model (e.g., GPT-4) or an external knowledge base to assess factual correctness that may not always be accessible. Addressing this, we propose Atomic Consistency Preference Optimization (ACPO), a self-supervised preference-tuning method that enhances factual accuracy without external supervision. ACPO leverages atomic consistency signals (i.e., the agreement of individual facts across multiple stochastic responses) to identify high- and low-quality data pairs for model alignment. Despite being fully self-supervised, ACPO outperforms the strong supervised alignment baseline by 1.95 points averaged across Phi-3 and Llama3 on the LongFact and BioGen datasets, demonstrating its effectiveness in improving factual reliability without relying on external models or knowledge bases.

1 Introduction

Large Language Models (LLMs) have emerged as powerful tools for accessing information through natural language generation. Long-form factoid question-answering (QA), in particular, plays a crucial role in human interactions with LLMs for information retrieval (AlKhamissi et al., 2022). However, a significant concern with LLMs is their tendency to produce content that appears plausible but is factually incorrect, a phenomenon commonly referred to as hallucination (Rawte et al., 2023; Xu et al., 2024; Huang et al., 2025). This issue is especially critical in the use of LLMs in domains like medical diagnosing, news reporting, and educational tutoring. To mitigate this issue, numerous strategies have been proposed.

The most common way to mitigate hallucinations involves a model alignment step to improve its factual accuracy. This process leverages curated (factual, non-factual) data pairs to align the model toward generating more factual content (Zhang et al., 2024; Tian et al., 2023). Typically, these data pairs are identified using a retriever paired with a knowledge base or a more advanced language model (like GPT-4) (Huang and Chen, 2024; Zhang et al., 2024). However, the applicability of these techniques is often limited by two key factors.

First, the unavailability of robust structured knowledge bases in many scenarios, particularly in low-resource domains such as IT technical support (Yang et al., 2023), medicine, and law (Sengupta et al., 2025) restricts the effectiveness of these methods. Second, relying on advanced proprietary APIs (e.g., GPT-4 or Gemini 2.5) to score alignment data is very expensive. It also introduces serious privacy risks, especially when scored data includes sensitive information. These challenges underscore the need for self-supervised approaches that can enhance factual accuracy and reduce hallucinations without relying on knowledge bases or external models.

Conversely, several inference-time techniques, such as ASC (Thirukovalluru et al., 2024), CoVe (Dhuliawala et al., 2024), and USC (Chen et al., 2024), operate in a fully self-supervised manner (not relying on any external models/knowledge bases) by evaluating an LLM's output using *self-consistency* or *self-evaluation* based mechanisms. These methods are designed to identify and eliminate non-factual components, delivering a more reliable final response. However, they are computationally intensive at inference time, often necessitating multiple LLM calls (e.g., verifying each individual fact) to improve performance.

Inspired by the success of these inference-time hallucination reduction techniques in long-form question answering, we propose a self-supervised

* Equal contribution.

Proceedings of the 14th International Joint Conference on Natural Language Processing and the 4th Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics, pages 1951–1963

training approach that extends these principles to the alignment phase, enabling models to learn inherent factuality without reliance on external supervision. Although versions of self-supervised preference tuning has been explored in prior works, such as FactTune and SKT (Zhang et al., 2024; Tian et al., 2023), these methods remain computationally expensive due to their heavy reliance on GPT-3.5 for extracting individual atomic facts and verification questions. Further, for factual calibration, self-consistency-based approaches have been shown to outperform confidence estimation methods such as self-evaluation in SKT and atomic question confidence in FactTune (Huang et al., 2024). Recent work on reasoning tasks further highlights that self-consistency-based alignment tuning outperforms alternative methods (Prasad et al., 2024).

We propose Atomic Consistency Preference Optimization (ACPO), a scalable self-supervised framework for enhancing factuality in long-form generation. Unlike prior methods, ACPO does not depend on external knowledge bases or stronger models. Instead, it relies solely on a base large language model and a lightweight BERT-based embedder. Specifically, ACPO applies atomic self-consistency—the factual agreement across multiple stochastically sampled responses (Thirukovalluru et al., 2024)—to efficiently construct preference-alignment pairs without supervision. Our contributions are:

- ACPO a novel, privacy-guaranteed, cost-efficient self-supervised preference tuning method to improve long-form factoid QA abilities without reliance on any stronger LLMs or knowledge bases.
- ACPO effectively reduces hallucinations and outperforms FactAlign, a strong supervised baseline, by improving factual precision by an average of $+1.95$ points on Phi-3 and Llama3 over fact-checking benchmarks: LongFact (Wei et al., 2024) and BioGen (Min et al., 2023).
- Through systematic ablations, we show that atomic self-consistency provides a strong and effective signal for the reinforcement learning step of large language models, outperforming its direct application at inference time.

2 Related Work

This section provides a comprehensive overview of inference-time methods and preference-tuning approaches aimed at reducing hallucinations and improving long-form question answering.

2.1 Inference Time Methods

2.1.1 Using Retrievers, Self-Evaluation

FactScore (Min et al., 2023) uses an external retriever to evaluate and improve response factuality. LongFact (Wei et al., 2024) extends the original FactScore metric by incorporating an F1-based evaluation for recall level factual assessment. Chain of Verification (CoVe) (Dhuliawala et al., 2024) introduces a method that generates multiple verification questions for a given response, retaining only the segments that can be independently verified. Similarly, Agrawal et al. (2024) filters non-factual content from list-style answers using indirect self-evaluation questions.

2.1.2 Using Self-Consistency

Consistency across stochastic responses has been proven to be a strong signal for improving reasoning and code generation (Chen et al., 2024; Wang et al., 2023). Building on this, SelfCheckGPT (Manakul et al., 2023) uses agreement among diverse model outputs as an indicator of hallucination. HaLo (Elaraby et al., 2023) used consistency-based metrics to detect sentence-level hallucinations in the generations. Atomic Self-Consistency (ASC) (Thirukovalluru et al., 2024) extends consistency-based methods by decomposing multiple stochastic responses into atomic facts, clustering them to reduce redundancy, and using cluster strength as a proxy for factual consistency. Inspired by ASC, we leverage atomic-level consistency signals to construct preference pairs for alignment.

2.2 Alignment Methods

Although inference-time methods have proven effective in reducing hallucinations, they are often computationally expensive, typically relying on multiple stochastic generations or repeated LLM queries to verify individual atomic facts within a response. To address this, recent work has focused on alignment-based training approaches that aim to induce factuality during training thus reducing the inference-time costs.

FactAlign (Huang and Chen, 2024), a strong supervised baseline, leverages Kahneman-Tversky Optimization (KTO) to align models using atomic fact labels from FactScore, which identifies individual facts using GPT-3.5 models and verifies them via a Wiki-based retriever.

SKT (Zhang et al., 2024) uses GPT-3.5-models to first generate atomic facts and then verifying

questions from multiple stochastic responses. An external retriever is then used to score each atomic fact, with scores aggregated to produce response-level ratings. Similarly, FactTune (Tian et al., 2023) generates atomic claims and corresponding questions using GPT-3.5, and scores them with an external retriever, aggregating claim-level scores to obtain response-level scores. In both methods, these scores are used to construct preference pairs—high-scoring responses as preferred and low-scoring as non-preferred. These pairs are then used for DPO-based alignment training of the model.

SKT and FactTune also propose self-supervised variants of their methods, wherein the base model is directly used to score atomic factuality instead of relying on an external retriever. However, these variants are *not truly self-supervised*, as they still depend on an additional GPT-3.5-based pipeline to generate atomic claims and verification questions. Assuming m stochastic responses are generated, both methods require m base LLM calls for the initial generations. This is followed by approximately $(m \times f \times 2)$ GPT-3.5 calls for generating f atomic claims and verification questions per response. Finally, the self-evaluation scores are computed using an additional $(k \times m \times f)$ base LLM calls, where $k \approx 1$ for SKT and $k \approx 20$ for FactTune—making the overall process extremely expensive. FactTune (Tian et al., 2023) also offers a GPT-3.5–free variant using entity recognizers for fact extraction, but it performs notably worse than the GPT-3.5-based version.

In terms of scoring, SKT leverages the generated atomic claims to estimate self-evaluation scores, while FactTune disregards the claims entirely and bases its confidence estimation solely on the calibration of atomic questions to estimate a response confidence. Notably, for factual calibration, self-consistency-based approaches have been shown to outperform self-evaluation-based scoring—as used in SKT—and other methods like atomic question confidence, as used in FactTune (Huang et al., 2024). Recent work shows that self-consistency–driven preference tuning significantly outperforms other baselines on reasoning tasks (Prasad et al., 2024).

Motivated by these findings, we propose Atomic Consistency Preference Optimization (ACPO), which leverages atomic self-consistency—the agreement of individual facts across stochastic responses—to score outputs and construct preference data for DPO-based alignment. ACPO generates m stochastic responses using only m base-LLM calls and eliminates the need for costly atomic fact labeling or large-model verification (e.g., GPT-3.5). Instead, it employs a lightweight embedding model to cluster atomic facts and uses cluster strengths as a measure of consistency. This design substantially improves efficiency while fostering strong factual consistency. Section 3 discusses basics of DPO followed by our methodology in Section 4.

3 Background: DPO Alignment Mechanism

Reinforcement Learning with Human Feedback (RLHF) has become a foundational approach for aligning large language models (LLMs) with human preferences and reducing hallucinations (Tian et al., 2023; Zhang et al., 2024). This line of work began with InstructGPT, which introduced a reward model and Proximal Policy Optimization (PPO) for fine-tuning (Ouyang et al., 2022). To reduce the cost of human annotations and leverage the growing capabilities of LLMs, later approaches such as Constitutional AI (Bai et al., 2022) and RLAIF (Lee et al., 2024) replaced human preferences with model-generated critiques. More recently, Direct Preference Optimization (DPO) (Rafailov et al., 2023) simplified this process by eliminating the need for a separate reward model and complex reinforcement learning, instead directly optimizing log-likelihood ratios over preference pairs. In this work, we create self-supervised preference data and adopt DPO to perform alignment tuning.

We apply the standard DPO loss function, shown in Equation 1.

$$\mathcal{L}_{\text{DPO}}(\pi_\theta; \pi_{\text{ref}}) = -\mathbb{E}_{(x,y_w,y_l)\sim\mathcal{D}}\Bigg[\log \sigma\Big(\beta \log \frac{\pi_\theta(y_w|x)}{\pi_{\text{ref}}(y_w|x)}\Big) - \log \sigma\Big(\beta \log \frac{\pi_\theta(y_l|x)}{\pi_{\text{ref}}(y_l|x)}\Big)\Bigg] \tag{1}$$

The DPO approach fine-tunes a policy π_θ by maximizing the preference margin between a preferred response y_w and a less preferred one y_l, relative to a reference policy π_{ref}. The β controls how aggressively the model separates preferred from non-preferred responses.

4 Methodology

In this section, we present our ACPO framework, detailing the training data generation process and the fine-tuning methodology.

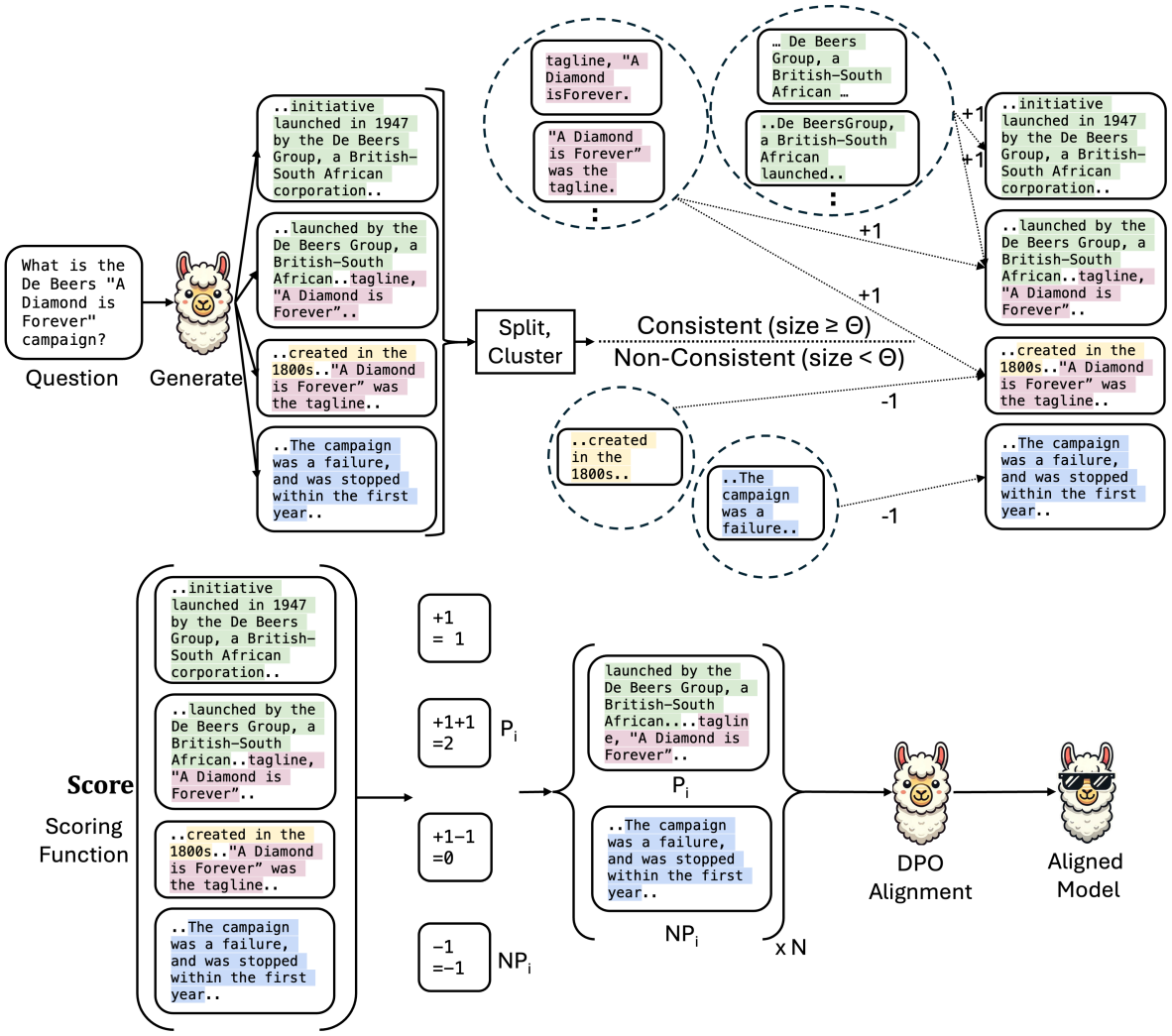

Figure 1: ACPO data curation pipeline. Steps 1–5 (Top): Generate stochastic responses for a question, Split and extract atomic facts; Cluster the atomic facts; Identify consistent and non-consistent clusters; Score responses based on cluster consistency. Step 6-7 (Bottom): Highest and lowest scoring curated as preference pairs; DPO alignment.

4.1 Overview

We leverage the model's generation capabilities to produce m stochastic responses for a given prompt P. Following the ASC (Thirukovalluru et al., 2024) framework, each response R_i is decomposed into a set of atomic facts $[a_1, a_2 \ldots a_k]$. All atomic facts from all responses are then aggregated and clustered. The core idea from ASC is that atomic facts appearing in larger clusters are more likely to be factual; we refer to these as consistent clusters $\mathcal{C}_i$, while smaller clusters form Non-consistent clusters $\mathcal{NC}_i$. For each atomic fact in a response R_i, we determine whether it belongs to a consistent or non-consistent cluster. If the atomic fact belongs to a consistent cluster, we reward its initial response (by adding a positive score); if not, we apply a penalty. This results in a consistency-based score for each response R_i, allowing us to distinguish between preferred and non-preferred responses. Next, we describe the scoring mechanism and training data generation for DPO alignment in detail.

4.2 Data Generation

4.2.1 Step 1: Initial Responses Generation

Given a question q, our objective is to prompt a large language model L to generate a response that is both accurate and informative. To achieve this, we adapt the system prompt from FactTune (Tian et al., 2023), modifying it to: *"You are an intelligent assistant who answers questions accurately"*. This modified prompt is then concatenated with the input of the actual question q. As a result, we obtain m independent responses denoted as $[R_1, R_2, \ldots, R_i, \ldots, R_m]$ by querying the model L with q using the predefined prompt P.

4.2.2 Step 2: Splitting Initial Responses for Atomic Facts

We then decompose each candidate response into a set of atomic statements. A single response R_i to a question may contain multiple sentences, each potentially expressing one or more atomic facts. While prior work (Min et al., 2023; Zhang et al., 2024; Huang and Chen, 2024) has employed large instruction-tuned models, like GPT-3.5, to identify atomic facts from long-form text, these methods are often computationally expensive and lack scalability. Inspired by Arslan et al. (2020); Thirukovalluru et al. (2024); Liu et al. (2023), we adopt a simplified yet effective alternative: treating each sentence in a generation as an atomic fact. Specifically, following the ASC paper, we apply standard sentence tokenization techniques (Bird et al., 2009) to segment each response into individual sentences, which we regard as atomic facts. After tokenization, the i-th response R_i is represented as a list of atomic sentences (atomic facts) $[a_{i1}, a_{i2}, \ldots, a_{ik}]$, where k is the atomic fact count. Note that although ACPO uses sentences as atomic facts, it is compatible with any atomic fact identification method (e.g., GPT-4), after which the subsequent steps can be applied.

4.2.3 Step 3: Clustering Atomic Facts

To address the high computational cost of verifying the relevance of each atomic fact across multiple generated responses, we follow the ASC framework by clustering semantically similar atomic units. ASC applies agglomerative clustering on sentence embeddings obtained from SimCSE (Gao et al., 2021) (a lightweight BERT-based sentence embedder), leveraging the substantial semantic overlap across generations to group atomic facts with similar meanings. Although agglomerative clustering has cubic worst-case complexity, it remains substantially more efficient than knowledge-base or LLM-based verification for each atomic fact.

4.2.4 Step 4: Consistent ($\mathcal{C}$) and Non-consistent ($\mathcal{NC}$) Clusters

Our method leverages the inherent consistency of model outputs that are quantified by the size of each cluster. Clusters with count below a threshold Θ are determined as $\mathcal{NC}_i$, while those above or equal to the Θ are classified as $\mathcal{C}_i$. The hypothesis is that LLMs are knowledgeable, and the high-frequency information in responses is more factual than rare ones (Wang et al., 2024). Therefore, information in $\mathcal{C}_i$ is more factual, and we utilize this property to score the initial responses R.

4.2.5 Step 5: Scoring Function

We define a consistency-based scoring function for each response R_i based on the classification of its atomic facts into consistent and non-consistent clusters. Let the atomic facts extracted from response R_i be: $R_i = [a_{i1}, a_{i2}, \ldots, a_{ik}]$. Let $\mathcal{C}_i$ denote the set of consistent clusters (with size $\geq \Theta$), and $\mathcal{NC}_i$ denote the set of non-consistent clusters (with size $< \Theta$). We score each response R_i as:

$$\text{Score}(R_i) = \sum_{j=1}^{k} \delta(a_{ij}); \quad \text{where}$$
$$\delta(a_{ij}) = \begin{cases} +1, & \text{if } a_{ij} \in \mathcal{C}_i \\ -1, & \text{if } a_{ij} \in \mathcal{NC}_i \\ 0, & \text{otherwise} \end{cases} \tag{2}$$

This scoring mechanism rewards atomic facts belonging to consistent clusters and penalizes those from non-consistent clusters. For example, if $R_i = [a_{i1}, a_{i2}, a_{i3}]$, and $a_{i1}, a_{i3} \in \mathcal{C}_i$ while $a_{i2} \in \mathcal{NC}_i$, then: $\text{Score}(R_i) = 1 + (-1) + 1 = 1$.

4.2.6 Step 6: Preference Data Obtain

After Steps 1-5, each R_i is assigned a consistency-based score. To construct preference pairs, we sort all responses by their scores and select the top-1 response as preferred (P_i) and the bottom-1 response as non-preferred (NP_i). This results in a training dataset: $\mathcal{D} = \{(x_i, P_i, NP_i)\}$, containing $|\mathcal{D}|$ (dataset size) datapoints where x_i is the prompt.

4.3 Step 7: DPO Alignment

The preference data pairs generated in § 4.2 are subsequently used for DPO alignment, with the detailed training setup described in § A.1.

5 Experiments

5.1 Models and Baselines

We conduct a comprehensive comparison of our proposed method, `ACPO`, against two key baselines. The first is `FactAlign` (Huang and Chen, 2024), a recently introduced alignment technique that leverages fine-grained, atomic fact-level annotations provided by the FactScore benchmark to guide the alignment process. The second baseline is the unaligned model, `RawModel`, which serves as a reference point to assess the impact of alignment strategies. To ensure a fair and thorough evaluation, we

Method	Llama-3-8B-Instruct				Phi-3-mini-4k-instruct			
	LongFact		BioGen		LongFact		BioGen	
	Score	#Claim	Score	#Claim	Score	#Claim	Score	#Claim
RawModel	79.8	121.2	55.9	61	78	90.2	41.7	88.4
FactAlign	**83.3**	119.9	57.1	56.7	81.2	113.8	47.1	100.9
ACPO	82.1	143.8	**58**	68	**84.6**	70.7	**51.8**	67.1

Table 1: Factscore accuracy for ACPO FactAlign and other baselines on LongFact, Bios datasets. FactAlign and ACPO were trained on the train set of LongFact. #Claim is the average number of claims produced by the model.

perform experiments at two different model scales: Phi-3 Mini (4B) and Llama3 (8B), allowing us to assess performance across varying levels of model capacity. ACPO uses $m = 30$ following Zhang et al. (2024). β is set to 0.1 during ACPO training. More details in §A.1. We are unable to compare with SKT due to unavailable code, and with FactTune as it requires 2M GPT-3.5 calls on LongFact dataset and uses private datasets during their training, preventing cross-evaluation.

5.2 Datasets and Evaluation

The training split of the LongFact dataset (Wei et al., 2024), consisting of 2,097 examples, was used to align both FactAlign and the proposed model, ACPO. Evaluation was conducted on the test splits of the LongFact and BioGen datasets (Min et al., 2023), with results reported for all models and baselines. Both LongFact and BioGen are English-language datasets. As FactScore relies on topic names—which are not available for LongFact—GTR-XL was used to retrieve the most relevant documents from the full Wikipedia corpus to support factual grounding during evaluation. FactScore reports two key metrics: the factual precision of the claims present in the output and the total number of factual claims identified in the output. The former is the more important metric.

5.3 Main Results

Table 1 shows the comparison between ACPO and other methods. Despite not relying on any external signals like FactAlign, ACPO outperforms it in three of the four settings, achieving an average gain of $+1.95$ points on Phi-3 and Llama3 across the LongFact and BioGen benchmarks. With Llama3-8B, ACPO also produces outputs containing a greater number of claims. This stems from its use of the ASC principle to identify preferred and non-preferred responses for training. As noted by Huang et al. (2024), models vary in their calibration (i.e., consistency across stochastic responses), which we believe explains why Phi-3 generates fewer claims.

5.3.1 Which Models Benefit Most from ACPO?

We used a very small threshold (Θ=2) for identifying consistent clusters for both Phi-3 and Llama3. Note that ACPO leverages the confidence of a model in generated responses to pick preferred/non-preferred data. Not all models are equally calibrated (confident about their responses). Some models are more calibrated than others (Huang et al., 2024). Hence, some models might perform better with ACPO than others.

Table 2 shows the ratio of the number of consistent clusters and non-consistent ones. Despite the same small threshold (Θ), we note that Phi-3 has a much higher number of non-consistent clusters. Fewer consistent clusters suggest that the model is less calibrated compared to Llama3. Hence, Phi-3 tends to pick smaller responses as preferred ones (this can avoid the negative score from non-consistent clusters). As shown in Table 3, one can relax the constraint from ACPO to balance the length of the training dataset.

Training Data	(#Consistent: #Non-Consistent)
Phi-3-mini-4k-instruct	(1:3)
Llama-3-8B-Instruct	(1:1.8)

Table 2: Ratio of consistent to non-consistent data across training sets.

5.4 Analysis 1: Can Length Balancing Alignment data help?

Our analysis of the alignment training data revealed distinct trends in preference behavior across the Phi-3 and Llama3 datasets. Specifically, in Phi-3, preferred responses were generally shorter than

Method	LLama-3-8b-Instruct						Phi-3-mini-4k-instruct					
	Length		LongFact		BioGen		Length		LongFact		BioGen	
	P	NP	Score	#C	Score	#C	P	NP	Score	#C	Score	#C
ACPO	478	457	**82.1**	143.8	58	68	307	327	84.6	70.7	**51.8**	67.1
ACPO (5,5)	466	449	79.6	150.6	57.3	83.2	287	295	85.1	72.5	49.4	64.3
ACPO (5,4+1)	466	461	80.2	140.2	58.2	73.9	287	285	**85.4**	75.6	50.5	65.4
ACPO (5,3+2)	466	468	81.4	127.8	**59.7**	68	287	279	84.9	75.2	50.6	67.1

Table 3: Using length as an additional criterion to balance preferred and non-preferred data leads improves #Claims, thereby increasing recall. Results are better for Llama3 on BioGen and for Phi-3 on LongFact.

Method	F1	#Claim
RawModel	75.88	100.85
FactAlign	80.32	103.33
ACPO (Ours)	**83.84**	**120.17**

Table 4: Longfact F1 score with Llama-3-8b-Instruct

Model	Phi-3-mini-4k-instruct			
			BioGen	
	Inf.	Score(R_i)	Score	#Claims
RawModel	Direct	-5.61	42.9	85.3
	ASC	4.10	45.7	84.5
ACPO (It. 1)	Direct	1.27	51.8	67.1
	ASC	7.97	**54**	70.7
ACPO (It. 2)	Direct	2.30	48.2	80.5
	ASC	10.83	51.8	93.4

Table 5: Performance with ASC at inference time. ACPO with direct inference outperforms RawModel, showing the importance of ASC in training. Applying ASC at inference further boosts performance, with Iteration 1 performing best. Later iterations add no gains but still beat RawModel while generating more #Claims, highlighting ASC's value for alignment data.

non-preferred ones, whereas in Llama3, the opposite trend was observed—preferred responses tended to be longer. To evaluate whether explicitly incorporating response length into the training signal could enhance the performance of ACPO, we explored several variants that adjusted the length of non-preferred responses in alignment with these trends—shortening them for Phi-3 and lengthening them for Llama3.

Specifically, ACPO selects the single highest- and lowest-scoring responses as the preferred and non-preferred examples, respectively. In contrast, ACPO (5,5) expands this selection to the top five and bottom five responses while keeping the number of training steps unchanged. Building on this variant, we introduced a length-based modification,

Stages \ Methods	ACPO	Sentence +GPT-4	GPT-4 (SKT)
Embedding	2.9 s	-	-
Clustering	0.39 s	-	-
Generating Atomic Facts	-	-	594.9 s
Verifying Atomic Facts	-	922.1 s	781.1 s
Total	**3.3 s**	922.1 s	1376.0 s

Table 6: Comparison of computational time across different methods for preference pair construction. Our ACPO employs a clustering-based approach, whereas Sentence+GPT-4 treats sentences as atomic facts and verifies them using GPT-4, and GPT-4 (SKT) both generates and verifies atomic facts through GPT-4.

replacing one or two of the non-preferred responses with alternatives chosen by length—favoring longer responses for Phi-3 and shorter ones for Llama3. As shown in Table 3, this adjustment yields performance improvements. Additionally, length balancing increases the #Claims, which is valuable in scenarios prioritizing high recall. This is because ACPO's selection process does not constrain response length, while the factual precision metric is length-sensitive—short responses can inflate precision scores (Huang and Chen, 2024).

5.5 Analysis 2: Measuring Recall

While Table 1 reports results for factual precision, recall is also critical in certain scenarios. Therefore, we additionally compute the F1 score of different models under LongFact using their custom API. It is important to note that FactScore does not provide an F1 metric; hence, we do not report it. As shown in Table 4, ACPO outperforms other methods.

5.6 Analysis 3: ACPO's Efficiency in Preference Pair Construction

Compared to LLM-based alignment approaches, ACPO achieves a substantial improvement in computational efficiency. As reported in Table 6, ACPO

processes and constructs preference data in just 3.3 seconds per example (2.9 s for embedding and 0.39 s for clustering), whereas *Using Sentences as Atomic Facts and Verifying Them Using GPT-4* requires 922.1 s, and *Generating Atomic Facts Using GPT-4 and Verifying Them Using GPT-4 (SKT)* requires 1376.0 s. This corresponds to an efficiency gain exceeding two orders of magnitude, primarily due to ACPO's fully self-supervised design, which eliminates the need for factual verification via large language models.

5.7 Ablation 1: ACPO compared with Inference time ASC

This ablation study investigates whether explicit alignment training is necessary or if the ASC principle can be effectively applied directly at inference time to achieve performance comparable to or better than ACPO. Although applying ASC at inference time is computationally more expensive, we assess its effectiveness relative to aligned models. Results are shown in Table 5. Direct refers to generating responses from the raw or trained model without any additional sampling, whereas ASC generation involves sampling multiple stochastic outputs and selecting the highest-scoring one based on Score(R_i). As observed, ACPO with direct decoding outperforms ASC applied to the unaligned RawModel, suggesting that incorporating ASC into the training process to construct a preference dataset leads to more substantial improvements than applying it only at inference time. Moreover, applying the ASC on top of ACPO yields further gains. Motivated by the improvements from ACPO+ASC, we conducted an additional round of self-supervised training using the already-aligned model. But this attempt did not improve test performance—likely due to overfitting after 25 epochs of ACPO training (§A.1). Iteration 2 had a higher Score(R_i) score than Iteration 1, which suggests that, although the model became more internally consistent, the improvement did not generalize to the test set, likely due to overfitting on the LongFact training data.

5.8 Ablation 2: Dissecting ACPO Scoring Mechanism

To understand the contribution of individual components within ACPO, we conduct a series of studies, with results summarized in Table 7. In its full form, ACPO rewards responses that include atomic facts from consistent clusters and penalizes those containing atomic facts from non-consistent ones.

Method	BioGen Score	BioGen #Claim
ACPO	51.8	67.1
ACPO (5,5)	49.4	64.3
ACPO w/o NC Penalty	46.9	99.7
Longest Preferred	41.3	97.6
Shortest Preferred	42.8	10.6

Table 7: Stronger preference signals in ACPO perform better than weaker ones in ACPO(5,5). Not penalizing non-consistent atomic facts yields worse alignment. Favoring short or long responses harms factual precision, highlighting the value of ACPO's alignment strategy.

We examine ACPO (5,5) alongside ACPO w/o Non-Consistent Penalty, which removes the penalty for selecting atomic facts from non-consistent clusters. To probe the effect of response length on alignment, we further extend our ablation by explicitly preferring either the longest or shortest responses, using a randomly selected response as the negative in both cases. Without the penalty, the model tends to favor longer responses, often ranking them more highly. The similarity in the number of factual claims between the penalty-free variant and Longest Preferred suggests that removing the penalty encourages behavior akin to explicitly favoring longer responses, underscoring the penalty's role in guiding alignment toward more precise and reliable outputs. Notably, the Shortest Preferred variant produces very few factual claims and performs poorly on FactScore precision, indicating that simply favoring brevity is not an effective strategy for improving factual alignment.

Method	Score	#Claims
ACPO($\Theta = 2$)	51.8	67.1
ACPO($\Theta = 3$)	50.4	48.7

Table 8: ACPO performance under different Θ values.

Temp.	0	0.3	0.6	0.9	1.2	1.5	ACPO
Score	42.9	43.2	41.7	38.4	36.3	33.3	**51.8**
#Claim	85.3	85.2	88.4	93	99.9	111	67.1

Table 9: Self-supervised alignment in ACPO (temp=0) outperforms temperature scaling—a simple post-hoc calibration method—across multiple temperatures.

5.9 Ablation 3: Effect of Θ

A higher value of Θ encourages the model to favor highly consistent responses, which often leads to

Question | "What is the De Beers diamond monopoly controversy?	
Preferred	**Non-Preferred**
The De Beers diamond monopoly controversy centers ... by the De Beers company, ... had a near-complete monopoly over the global diamond market. Several key details and examples include: 1. De Beers' monopoly began in the late 19th century when Cecil Rhodes, a British businessman and mining magnate, founded the company in South Africa in 1888. 2. By the early 20th century, De Beers ... and controlled over 90% of the world's rough diamond production and trade. 3. The company's ... stockpiling diamonds and controlling ... to maintain high prices and ensure a stable supply. ... 7. In 2005, the Antitrust Division of the United States Department of Justice filed an antitrust lawsuit ... price-fixing and maintaining a monopoly in the diamond market....	The De Beers diamond monopoly controversy revolves around the company's control over the global diamond market in the ... early 21st centuries. De Beers, a British company founded in 1888 by Cecil Rhodes, came to dominate the diamond industry by controlling diamond production and distribution. ... De Beers' monopoly was challenged by a South African-based company, Cut Stone, led by Ernest Oppenheimer, grandson of ..., Cecil Rhodes. Cut Stone accused De Beers of fixing diamond prices and controlling the market. The controversy intensified when the South African government, ..., announced plans to nationalize the diamond industry in 1981. ... In 1998, a class action lawsuit was filed against De Beers in the United States by the American Antitrust Institute (AAI)....

Table 10: Preferred and Non-Preferred responses curated by ACPO. Green highlights indicate phrases verified as correct, while red highlights mark incorrect ones according to Wikipedia. In this example, terms like Cutstone, the nationalization claim, AAI are hallucinated.

the selection of shorter responses, as maintaining consistency is easier with fewer facts. Results are shown in Table 8.

5.10 Analysis 4: Can simple calibration techniques match ACPO performance?

ACPO is a simple self-supervised algorithm that uses the atomic consistency principle to align the model to generate better responses. Temperature scaling is another way of model calibration (Renze, 2024). We investigate if the gains in ACPO can be achieved by simple temperature scaling of the RawModel. Table 9 shows the results. ACPO significantly outperforms all temperature settings of the RawModel.

5.11 Analysis 5: Qualitative Analysis

Preferred and non-preferred examples curated by ACPO are shown in Table 10. As the highlights indicate, ACPO identifies high-quality examples without relying on external signals.

6 Conclusion

We introduce Atomic Consistency Preference Optimization (ACPO), a self-supervised method for aligning LLMs to improve factual accuracy in long-form question answering. ACPO leverages the atomic self-consistency principle to curate high-quality preference data, eliminating the need for external supervision or strong LLMs. By identifying preferred and non-preferred generations based on internal consistency signals, ACPO enables efficient and scalable DPO training. Our extensive evaluations on LongFact and BioGen show that ACPO not only outperforms a strong supervised baseline (FactAlign), but also surpasses all temperature-tuned variants of unaligned models. Furthermore, we show that using atomic consistency during training leads to better factual precision than applying it solely at inference time. Additional ablations validate the length penalties and the robustness of ACPO across different model sizes. In summary, ACPO presents a simple, effective, and efficient self-supervised approach to enhance factual alignment in LLMs, leading to more trustworthy and factual generation capabilities.

7 Limitations

The self-consistency principles employed in this work present opportunities for integration with self-evaluation strategies, potentially enabling the development of hybrid self-supervised alignment frameworks that combine the strengths of both paradigms. Such approaches could leverage self-consistency for generating reliable preference signals while incorporating self-evaluation mechanisms to further refine alignment quality. However, in this study, we deliberately focus on self-consistency-based methods to isolate and rigorously assess their effectiveness.

8 Ethics Statement

While our model is not tied to any specific applications, it could be used in sensitive contexts such as health-care, etc. Any work using our method is requested to undertake extensive quality-assurance and robustness testing before applying in their setting. To the best of our knowledge, datasets used in our work do not contain any sensitive information.

9 Reproducibility Statement

Code: https://github.com/JingfengSteven/ACPO
License: Datasets and methods utilized in this study are under the Apache License 2.0 or the MIT License. This research adheres to the respective licensing terms. Outputs of this work are released under the Apache License 2.0.

References

Ayush Agrawal, Mirac Suzgun, Lester Mackey, and Adam Kalai. 2024. Do language models know when they're hallucinating references? In *Findings of the Association for Computational Linguistics: EACL 2024*, pages 912–928, St. Julian's, Malta. Association for Computational Linguistics.

Badr AlKhamissi, Millicent Li, Asli Celikyilmaz, Mona T. Diab, and Marjan Ghazvininejad. 2022. A review on language models as knowledge bases. *CoRR*, abs/2204.06031.

Fatma Arslan, Naeemul Hassan, Chengkai Li, and Mark Tremayne. 2020. A benchmark dataset of check-worthy factual claims. *Proceedings of the International AAAI Conference on Web and Social Media*, 14(1):821–829.

Yuntao Bai, Saurav Kadavath, Sandipan Kundu, Amanda Askell, Jackson Kernion, Andy Jones, Anna Chen, Anna Goldie, Azalia Mirhoseini, Cameron McKinnon, and 1 others. 2022. Constitutional ai: Harmlessness from ai feedback. *arXiv preprint arXiv:2212.08073*.

Steven Bird, Ewan Klein, and Edward Loper. 2009. *Natural language processing with Python: analyzing text with the natural language toolkit*. " O'Reilly Media, Inc.".

Xinyun Chen, Renat Aksitov, Uri Alon, Jie Ren, Kefan Xiao, Pengcheng Yin, Sushant Prakash, Charles Sutton, Xuezhi Wang, and Denny Zhou. 2024. Universal self-consistency for large language models. In *ICML 2024 Workshop on In-Context Learning*.

Shehzaad Dhuliawala, Mojtaba Komeili, Jing Xu, Roberta Raileanu, Xian Li, Asli Celikyilmaz, and Jason Weston. 2024. Chain-of-verification reduces hallucination in large language models. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 3563–3578, Bangkok, Thailand. Association for Computational Linguistics.

Mohamed Elaraby, Mengyin Lu, Jacob Dunn, Xueying Zhang, Yu Wang, and Shizhu Liu. 2023. Halo: Estimation and reduction of hallucinations in open-source weak large language models. *CoRR*, abs/2308.11764.

Tianyu Gao, Xingcheng Yao, and Danqi Chen. 2021. SimCSE: Simple contrastive learning of sentence embeddings. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6894–6910, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Chao-Wei Huang and Yun-Nung Chen. 2024. FactAlign: Long-form factuality alignment of large language models. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 16363–16375, Miami, Florida, USA. Association for Computational Linguistics.

Lei Huang, Weijiang Yu, Weitao Ma, Weihong Zhong, Zhangyin Feng, Haotian Wang, Qianglong Chen, Weihua Peng, Xiaocheng Feng, Bing Qin, and Ting Liu. 2025. A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions. *ACM Trans. Inf. Syst.*, 43(2).

Yukun Huang, Yixin Liu, Raghuveer Thirukovalluru, Arman Cohan, and Bhuwan Dhingra. 2024. Calibrating long-form generations from large language models. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 13441–13460, Miami, Florida, USA. Association for Computational Linguistics.

Harrison Lee, Samrat Phatale, Hassan Mansoor, Thomas Mesnard, Johan Ferret, Kellie Lu, Colton Bishop, Ethan Hall, Victor Carbune, Abhinav Rastogi, and Sushant Prakash. 2024. Rlaif vs. rlhf: scaling reinforcement learning from human feedback with ai feedback. In *Proceedings of the 41st International Conference on Machine Learning*, ICML'24. JMLR.org.

Nelson Liu, Tianyi Zhang, and Percy Liang. 2023. Evaluating verifiability in generative search engines. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 7001–7025, Singapore. Association for Computational Linguistics.

Potsawee Manakul, Adian Liusie, and Mark Gales. 2023. SelfCheckGPT: Zero-resource black-box hallucination detection for generative large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 9004–9017, Singapore. 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*, Singapore.

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, John Schulman, Jacob Hilton, Fraser Kelton, Luke Miller, Maddie Simens, Amanda Askell, Peter Welinder, Paul Christiano, Jan Leike, and Ryan Lowe. 2022. Training language models to follow instructions with human feedback. In *Proceedings of the 36th International Conference on Neural Information Processing Systems*, NIPS '22, Red Hook, NY, USA. Curran Associates Inc.

Archiki Prasad, Weizhe Yuan, Richard Yuanzhe Pang, Jing Xu, Maryam Fazel-Zarandi, Mohit Bansal, Sainbayar Sukhbaatar, Jason Weston, and Jane Yu. 2024. Self-consistency preference optimization. *arXiv preprint arXiv:2411.04109*.

Rafael Rafailov, Archit Sharma, Eric Mitchell, Stefano Ermon, Christopher D. Manning, and Chelsea Finn.

2023. Direct preference optimization: your language model is secretly a reward model. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, NIPS '23, Red Hook, NY, USA. Curran Associates Inc.

Vipula Rawte, Swagata Chakraborty, Agnibh Pathak, Anubhav Sarkar, S.M Towhidul Islam Tonmoy, Aman Chadha, Amit Sheth, and Amitava Das. 2023. The troubling emergence of hallucination in large language models - an extensive definition, quantification, and prescriptive remediations. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 2541–2573, Singapore. Association for Computational Linguistics.

Matthew Renze. 2024. The effect of sampling temperature on problem solving in large language models. In *Findings of the Association for Computational Linguistics: EMNLP 2024*, pages 7346–7356, Miami, Florida, USA. Association for Computational Linguistics.

Saptarshi Sengupta, Wenpeng Yin, Preslav Nakov, Shreya Ghosh, and Suhang Wang. 2025. Exploring language model generalization in low-resource extractive QA. In *Proceedings of the 31st International Conference on Computational Linguistics*, pages 7106–7126, Abu Dhabi, UAE. Association for Computational Linguistics.

Raghuveer Thirukovalluru, Yukun Huang, and Bhuwan Dhingra. 2024. Atomic self-consistency for better long form generations. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 12681–12694, Miami, Florida, USA. Association for Computational Linguistics.

Katherine Tian, Eric Mitchell, Huaxiu Yao, Christopher Manning, and Chelsea Finn. 2023. Fine-tuning language models for factuality. In *NeurIPS 2023 Workshop on Instruction Tuning and Instruction Following*.

Ante Wang, Linfeng Song, Baolin Peng, Lifeng Jin, Ye Tian, Haitao Mi, Jinsong Su, and Dong Yu. 2024. Improving LLM generations via fine-grained self-endorsement. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 8424–8436, Bangkok, Thailand. Association for Computational Linguistics.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc V Le, Ed H. Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. 2023. Self-consistency improves chain of thought reasoning in language models. In *The Eleventh International Conference on Learning Representations*.

Jerry Wei, Chengrun Yang, Xinying Song, Yifeng Lu, Nathan Zixia Hu, Jie Huang, Dustin Tran, Daiyi Peng, Ruibo Liu, Da Huang, Cosmo Du, and Quoc V Le. 2024. Long-form factuality in large language models. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*.

Can Xu, Qingfeng Sun, Kai Zheng, Xiubo Geng, Pu Zhao, Jiazhan Feng, Chongyang Tao, Qingwei Lin, and Daxin Jiang. 2024. WizardLM: Empowering large pre-trained language models to follow complex instructions. In *The Twelfth International Conference on Learning Representations*.

Fangkai Yang, Pu Zhao, Zezhong Wang, Lu Wang, Bo Qiao, Jue Zhang, Mohit Garg, Qingwei Lin, Saravan Rajmohan, and Dongmei Zhang. 2023. Empower large language model to perform better on industrial domain-specific question answering. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing: Industry Track*, pages 294–312, Singapore. Association for Computational Linguistics.

Xiaoying Zhang, Baolin Peng, Ye Tian, Jingyan Zhou, Lifeng Jin, Linfeng Song, Haitao Mi, and Helen Meng. 2024. Self-alignment for factuality: Mitigating hallucinations in LLMs via self-evaluation. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1946–1965, Bangkok, Thailand. Association for Computational Linguistics.

A Appendix

A.1 Training Details

The alignment procedure is adapted from Tian et al. (2023). For training, we set the batch size to 32 for the Phi-3-mini-4k-instruct model and 64 for the LLaMA-3-8B-Instruct model. We use a linear warmup learning rate schedule, with 100 warmup steps for the LLaMA model and 150 for the Phi model, followed by cosine decay. The learning rate is kept as default=$1e^{-6}$, and β is set as 0.1. Rather than using a fixed number of epochs, training is controlled by the total number of steps. Given that we use either ACPO (5,5) $(5 \times 5 = 25)$ preference pairs per question or ACPO (1,1) $(1 \times 1 = 1)$ preference pairs per question, we ensure 1 complete epoch for the 25-pair case. Consequently, for the 1-pair case, we train for 25 epochs to maintain step parity. Gradient clipping is applied with a default threshold of 10. The total training time is approximately 1 hour for the Phi model and 2.5 hours for the LLaMA model, using 4 NVIDIA H800 GPUs with 80 GB of memory each.

We used the default temperature values - 0.5 for `FactAlign` (Huang and Chen, 2024) and 0.6 for `RawModel`. For `ACPO`, we use greedy decoding (temperature = 0) to ensure reproducibility and to evaluate the model's capability without introducing randomness. Results for greedy decoding of all models present in §11. `ACPO` beats `FactAlign` even in this setup.

A.2 Clustering Details

We employ Agglomerative Clustering with average linkage and cosine distance as the similarity metric. The number of clusters is determined dynamically by setting $n\_clusters = None$ and applying a $distance\_threshold = 0.15$, such that clusters are continuously merged until the pairwise inter-cluster distance exceeds the specified threshold. The Θ value for consistent and non-consistent filtering is set as 2.

A.3 Results with Greedy (temperature=0)

Table 11 shows the results for all models at temperature=0. `ACPO` beats baselines and `FactAlign` even at greedy decoding (temperature=0).

A.4 Data Sheet

We present the dataset details along with key statistics relevant to the clustering process in Table 12.

A.5 Data Generation, Training, Evaluation Prompts

The system prompt we use for the initial response generation is modified from FactTune (Tian et al., 2023), The User Prompt (questions) is kept the same as FactAlign (Huang and Chen, 2024).

Initial Response Generation (Training Data Creation)

System Prompt:
"You are an intelligent assistant who answers questions accurately."

User Prompt:
"What is the geographical importance of the Strait of Gibraltar? Provide as many specific details and examples as possible (such as names of people, numbers, events, locations, dates, times, etc)."

The training prompt for DPO alignment is kept as the default, which is the same question prompt as the generation part.

For the test data generation, The actual question exactly follows FactScore (Min et al., 2023) official repository:

Test Response Generation

System Prompt:
"You are an intelligent assistant who answers questions accurately."

User Prompt:
"Answer this question. Question: Tell me a bio of Kourosh Zolani."

Method	Llama-3-8B-Instruct				Phi-3-mini-4k-instruct			
	LongFact		BioGen		LongFact		BioGen	
	Score	#Claim	Score	#Claim	Score	#Claim	Score	#Claim
RawModel	79.5	121.6	55.3	61.1	79.8	91.1	42.9	85.3
FactAlign	**83.1**	118.2	57.6	58.1	82.6	112.2	48.4	97.3
ACPO	82.1	143.8	**58**	68	**84.6**	70.7	**51.8**	67.1

Table 11: ACPO vs other methods at temperature=0 during inference

Datasets	Model	(Train Test)	Response Number	ACS	ARC
LongFact	LLaMA-3	(2097,233)	30	290	23
	Phi-3.5-mini			245	14

Table 12: Summary of the training dataset statistics. Response Number denotes the number of initial responses generated per question. ACS (Average Cluster Size) represents the average number of clusters formed per question based on atomic fact clustering. ARC (Average Response Coverage) indicates the average number of clusters that each response contributes to