THE KNOWLEDGE IS ALREADY THERE: GEOMETRIC INTERVENTIONS FOR FACTUAL RECALL IN LLMS

Anonymous authors
Paper under double-blind review

ABSTRACT

Factual retrieval failures in large language models are often assumed to reflect missing knowledge. We investigate an alternative mode: *representational collapse*, in which questions requiring different answers occupy overlapping regions of the hidden-state space. We measure this by a question's *crowdedness*, the number of other questions with highly similar representations. Across three sizes of Qwen2.5, the more crowded half of TriviaQA questions is answered correctly 22–25 percentage points less often than the rest. We propose **C**ontrastive **R**epresentation **A**djustment for **F**ac**t**uality **(CRAFT)**, which reshapes this geometry through a contrastive objective over question paraphrases and negatives, without any answer supervision. Evaluated on TriviaQA, NQ-Open, and HotpotQA across three model families, CRAFT gives the best exact-match accuracy of all methods on every benchmark for all Qwen2.5 sizes, raising Qwen2.5-7B accuracy on TriviaQA from 38.4% to 58.2%. Finally, CRAFT largely removes representational collapse, consistent with many retrieval failures reflecting knowledge that is present in the model but hard to reach.

1 INTRODUCTION

Single-hop factual question answering remains a challenge for large language models (LLMs), which frequently output incorrect responses to questions requiring world knowledge. This phenomenon has two potential sources: the model either simply may not know the pertinent piece of information, or it may be failing to retrieve it, despite encoding it within its weights. A number of post-training and inference methodologies that address either of these failure modes have been proposed in the recent literature to address this challenge: for instance, probes that determine the availability of knowledge can help mitigate the first failure mode, while inference time or fine-tuning methods can steer the output towards correct retrieval.

Our work explores a particular way in which retrieval can fail, namely *representational collapse* in the model's hidden states. In simple terms, we say that such a collapse occurs whenever a question q is mapped to a region of the hidden-state space that is densely populated by other questions with different answers, that is, when many semantically distinct questions share similar internal representations. Such crowding can come from questions that are easily confused in meaning, but also from shared sentence structure or common question templates, even among questions that are otherwise unrelated. Since the generated answers are a function of these hidden representations, such crowdedness is associated with reduced reliability of the retrieved answers. By formalizing crowdedness through the number of highly-similar neighbouring representations, we first show that this crowdedness is prevalent, and is associated with lower accuracy. In particular, on Qwen2.5 models at all three sizes, EM accuracy on the more crowded half of questions is 22–25 percentage points lower than on the less crowded half, while overall base accuracy ranges from 21% to 38%.

This failure mode also naturally suggests a mitigation strategy: rather than attempting to expose the model to more facts, one may instead reshape representations to make information already present in the weights easier to retrieve. To this end, we propose a new unsupervised post-training method, called '**C**ontrastive **R**epresentation **A**djustment for **F**ac**t**uality' (CRAFT). As the name suggests, this method uses a contrastive objective to pull the hidden representations of semantically equivalent questions together, and push those of semantically distinct questions apart, thereby improving the geometric separation of semantically distinct concepts within the hidden representation space.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

Specifically, CRAFT uses a dataset of factual questions, without any need for accompanying answers. From these questions, we first construct an augmented dataset of "paraphrases" (semantically identical questions) and "negatives" (semantically distinct questions). We take the final-layer hidden state of the last token at the prompt boundary as the representation of a question. Subsequently, we optimise a standard contrastive loss over the representations of the triple of questions, paraphrases and negatives, along with a regularising repulsive term that limits representation collapse.

We evaluate CRAFT on three benchmarks on models with different scales. Against both inference-time and fine-tuning baselines, CRAFT delivers consistent accuracy gains and strong cross-dataset transferability (Table 1) without using any answer supervision in its training data. In greater detail, the key structural effect of CRAFT is precisely to reduce the extent of representation collapse. This improves accuracy broadly, for both previously crowded and uncrowded questions. For Qwen2.5-7B, EM accuracy on TriviaQA rises from 38.4% to 58.2%, which significantly outperforms prior unsupervised post-training and inference-time methods. Additionally, CRAFT is faster to train than prior pseudo-label based methods, and adds no inference time costs beyond the base model.

To summarise, our key contributions are thus:

- We identify representation collapse as a mode by which language models can fail to retrieve factual knowledge, and show that it is strongly associated with lower accuracy.
- We design a simple and efficient post-training strategy, CRAFT, that uses a contrastive objective to reduce the effect of representation collapse.
- We evaluate CRAFT, on multiple model sizes, architectures, and datasets, and find that it both significantly improves accuracy, and mitigates representation collapse. We further show that it improves accuracy at every level of crowdedness, and use ablations and controls to isolate the role of each design choice.

1.1 Related Work

Why are models not factual? LLMs encode vast factual knowledge but retrieve it unreliably (Ji et al., 2023; Wang et al., 2023), and often encode more than they express (Gekhman et al., 2025). Overlap between entity representations can also cause interference and hallucination during fine-tuning (Kaplan et al., 2026). Models often possess an internal sense of their own uncertainty yet still fail to surface the truth consistently (Kadavath et al., 2022). Recent theoretical work argues this is inherent to next-token prediction, which penalizes uncertainty and trains models to produce statistically plausible but factually baseless responses (Kalai et al., 2025). Entity-frequency biases further compound the problem: rare facts appear too infrequently during pre-training to form robust memories, causing models to confabulate plausible answers for obscure queries rather than correctly retrieve specific facts (Kandpal et al., 2023; Mallen et al., 2023).

Fine-tuning based methods: Post-training to improve factuality splits into supervised and unsupervised methods. The former includes methods like preference learning on factual data, integration of external search trajectories in training (Jin et al., 2025), and incorporating uncertainty into training (Xue et al., 2025). While effective, these methods rely on having access to answers, whereas our interest is in capturing answers already present in the base model weights. Thus, unsupervised methods are most pertinent. Typically, such methods operate by generating some pseudo-preference data from the questions, and applying an off-the-shelf fine tuning strategy. Examples include FactTune (Tian et al., 2024) and Self-Align methods (Zhang et al., 2024), which use distinct strategies to generate this data. In sharp contrast, our focus in on representation-level tuning, and we do not pursue any such pseudo-label generation.

Inference-time methods and representation engineering. Rather than updating model weights, inference-time approaches modify decoding or prompting (Cheng et al., 2025; Karan & Du, 2025). Contrastive decoding methods such as DoLa (Chuang et al., 2024) and DeCoRe (Gema et al., 2024) instead adjust logits across layers or attention heads. A parallel line of work steers model behavior through low-dimensional activation directions: ITI shifts activations along truth-correlated attention heads at generation time (Li et al., 2024), while locate-then-edit methods such as ROME and MEMIT implant corrected facts into feed-forward weights (Meng et al., 2022; 2023). In our experiments on Qwen2.5, DoLa and ITI often score below the base model.

2 SETUP: QUESTION REPRESENTATIONS, SIMILARITY, AND CORRECTNESS

The first part of this paper focuses on the presence and effect of representational collapse in language models, while the second describes CRAFT and evaluates its performance. This section describes how we capture the hidden representation of questions and measure similarity.

Representation of Questions. Given a question q, we format it as a prompt $p(q) =$ `"Question: {q}\nAnswer:"` and pass it through the model to obtain the hidden states at the last prompt token position. For a transformer model with parameters θ, we let $h_\ell^\theta(q) \in \mathbb{R}^d$ denote the hidden state at layer ℓ for question q, where $\ell \in \{0, 1, \ldots, L\}$ indexes the transformer layers including the embedding layer. The last prompt token attends to all preceding tokens under causal attention and therefore encodes a compressed representation of the full input, making it a natural choice for representing the question. We represent q by its final-layer hidden state,

$$\mathbf{H}_\theta(q) := h_L^\theta(q) \in \mathbb{R}^d. \tag{1}$$

We use the final layer for two reasons (§B.1). First, the association of crowdedness with both accuracy and next-token entropy grows sharply over roughly the last quarter of layers and is strongest at the final layer. Second, training CRAFT with final-layer similarity gives higher accuracy than using all layers. We use this choice for Qwen2.5 and SmolLM3. For Llama-2-7B, we instead use the all-layer similarity, which averages the per-layer cosine similarities, since it gave higher accuracy for this model (§B.2). In all cases, $\mathbf{H}_\theta$ uses only the final prompt token; no generated tokens are used to represent the question.

Similarity of Representations. The cosine similarity between two questions q and q' is defined as

$$\mathrm{sim}_\theta(q,\, q') \;=\; \frac{h_L^\theta(q)^\top h_L^\theta(q')}{\|h_L^\theta(q)\|\, \|h_L^\theta(q')\|}, \tag{2}$$

i.e., the cosine similarity between the final-layer representations of q and q'.

Evaluating Correctness. The main metric of interest in this paper is the accuracy of answering factual questions. In the factual question databases we evaluate models on, a question has a short list of valid answers. Throughout, we use greedy decoding of language models to generate the response to a question. Since these responses can be long, we measure correctness in two ways:

- **Exact Match (EM)**: a response is considered correct if it matches any valid answer exactly.
- **Contains Exact Match (C-EM)**: a response is considered correct if any valid answer appears within the *first sentence or first line* of the model's response, i.e., the text preceding the first sentence ending character (i.e. '`.`','`?`','`!`') or newline '`\n`' character in the output.

The EM criterion is evidently stricter than the C-EM criterion, since in this case the model may not produce further text beyond a valid answer. In C-EM, we restrict attention to the first sentence/line in order to penalize models which dump a large body of general information in response to a specific question. Containment criteria of this kind are used in prior work; for instance, Park et al. (2026) check containment within the first 32 generated tokens, which is more lenient towards long outputs than our first-sentence rule. EM remains the standard metric (Joshi et al., 2017), and we report both. **Accuracy** is simply computed as the fraction of questions that are answered correctly in the sense specified (i.e., EM or C-EM).

3 REPRESENTATIONAL COLLAPSE AS A FAILURE MODE

The main failure mode we investigate is that of 'representation collapse'. Informally, this is a geometric phenomenon whereby the hidden representations $\{\mathbf{H}_\theta(q)\}$s of question with distinct answers are very similar to one another. One mechanism by which such collapse may affect fact retrieval is self-evident: since the generation is driven by the hidden representations of the question, nearby questions with differing answers act as spoilers, and may cause retrieval of incorrect answers. A more subtle mechanism is that a model may respond to a question with many such nearby 'spoiler's with longer and, more 'hedged' answers (see also , §A.1), the net effect being a reduction in the probability mass placed on the correct answer. In either case, the more 'crowded' a region of the representation space is, the worse we expect the accuracy of questions in this region to be.

To formalize this, we need to specify what is meant by 'hidden representations being similar', and capture the degree of local crowding. To this end, for a **similarity threshold** $\tau \in [0, 1]$, let

$$\mathcal{C}_\tau^\theta(q) := \{q' : \mathrm{sim}_\theta(q, q') \geq \tau\}$$

denote the τ-neighbourhood of a question q. This τ is meant to denote a similarity level high enough that it has a functional effect on the downstream answer generation, and $\mathcal{C}_\tau^\theta(q)$ is the set of questions so close to q. Naturally, we use the size of $\mathcal{C}_\tau^\theta(q)$ as a **crowdedness score** of q, denoted $N_\tau^\theta(q) = |\mathcal{C}_\tau^\theta(q)|$. Effectively, we say that the model θ suffers representation collapse at q if $N_\tau^\theta(q)$ is large. Below, we will argue that representation collapse in this sense is prevalent in the base model, and we will use $\mathcal{C}_\tau^{\text{BASE}}$ and N_τ^{BASE} to refer to the neighbourhood and crowdedness in the base model.

3.1 Prevalence and Effect of Representation Collapse

We demonstrate the relevance of representation collapse to factual answering by establishing two facts for the base model:

- As the crowdedness of questions increases, the accuracy of answering them decreases, and
- Representation collapse occurs at a significant fraction of questions.

Figure 1 exhibits both phenomena on the Qwen2.5 base model with 1.5B, 3B, and 7B parameters (similar plots for Llama models and other datasets are in §A.4), using the de-duplicated TriviaQA validation set (Joshi et al., 2017) as the source of about $10K$ distinct questions.[1]

In this figure, we plot curves of EM-accuracy (see §2) averaged over questions with crowdedness scores in the $[\pi - 0.02, \pi]$'th percentile of crowdedness scores, where π is varied between 0 and 0.99. More explicitly, let κ_π denote the πth percentile of $\{N_\tau^{\text{BASE}}(q)\}_q$, and let $\mathcal{D}_\tau(\pi) := \{q : N_\tau^{\text{BASE}}(q) \in [\kappa_{\pi-0.02}, \kappa_\pi]\}$. Fig. 1 plots π against the accuracy of the model on questions in $\mathcal{D}_\tau(\pi)$. [2] The curves are presented for three choices of the similarity threshold τ, which are chosen as the 98th, 99th and 99.5th percentile of all similarity scores—for the 7B model, the 99th percentile is $\tau = 0.861$. Thus, we focus on crowdedness as defined by very high similarity levels, which are most pertinent for causing local confusion, and study the a wide range of crowdedness levels, thus capturing an overall sense of the effect of representation collapse on accuracy.

Observations. The first key observation is that in all curves, accuracy drops significantly, with the bin value π. In particular, we see a steady decline with the crowdedness percentile across a wide range, with further severe drops for extremely crowded (> 80 percentile of scores) questions. This explicitly demonstrates that crowdedness is strongly negatively associated with accuracy.

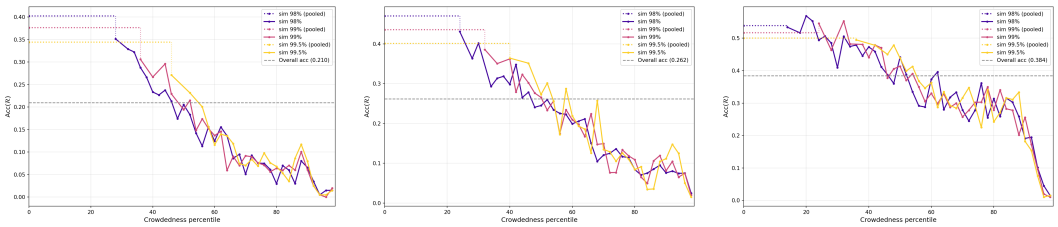
The second main observation is that this effect is pervasive. Accuracy drops below the overall base-model accuracy at around the median crowdedness level, meaning that a substantial fraction of questions lie in regions where retrieval is much less accurate. In particular, we note that accuracy over the top-half of crowdedness (i.e., over $\bigcup_{\pi \in [0.5,1]} \mathcal{D}_\tau(\pi)$) is 22–25 percentage points lower than that over the bottom half of crowdedness (see Fig. 5 in §A.2 for more details). This is especially significant considering that the base model accuracies range from 21–38% in the three cases studied. Percentiles are model-specific, so we also compare raw crowdedness at fixed thresholds (§B.7): it falls from 1.5B to 7B but rises again at 32B, so larger models are not uniformly less crowded, and the negative association with accuracy persists at 32B.

These observations show that representation collapse, as measured by N_τ^{BASE}, is strongly associated with a large decline in accuracy on the TriviaQA dataset. A key limitation is that this study is purely associative, with potentially uncontrolled confounders (e.g., the same plots could arise if model knowledge was negatively related to crowdedness). More convincing evidence would be yielded by an intervention that produces a change in accuracy by altering the degree of representation collapse.

CRAFT changes this geometry directly: it uses no answers, and simply tries to 'un-crowd' questions. In §5 we show that it reduces crowdedness and improves accuracy at every level of crowdedness. While not a causal study, this is consistent with the association.

[1] TriviaQA is a dataset of trivia questions with independently gathered answers. The questions are drawn from a broad set of domains. By 'de-duplicated,' we mean that repeated questions are removed.

[2] We chose this percentile-based presentation for two reasons: first, this directly captures the fraction of points affected at the particular crowdedness level, and second because the distribution of crowdedness varies over a wide scale



(a) 1.5B (b) 3B (c) 7B

Figure 1: Accuracy on the crowded set ($D_\tau(\pi)$) as a function of crowdedness percentile (π) for Qwen2.5 at 1.5B, 3B, and 7B parameters. Each curve corresponds to a different similarity floor τ. Observe that accuracy consistently drops as crowdedness rises, and in particular falls below the overall base-model accuracy for $\pi > 0.5$. Questions with more crowded representations are answered correctly less often, at every model size. The dotted horizontal segment at the left of each curve represents the pooled average accuracy over all questions with too few neighbours to form a resolved bin

4 Contrastive Representation Adjustment for Factuality

§3 shows that questions in dense neighbourhoods of the hidden-state space are answered correctly less often, at every model size. Can we intervene on this geometry directly to reduce crowding and recover accuracy?

Contrastive **R**epresentation **A**djustment for **F**ac**t**uality (CRAFT) is designed to do exactly this. Rather than training on correct answers, CRAFT reshapes the hidden-state trajectories of the model by pulling representations of semantically equivalent questions together while pushing representations of distinct questions apart, reducing crowdedness across the representation space. Crucially, this requires no answer supervision and introduces no changes to the model architecture.

CRAFT consists of two steps. First, we augment the training dataset to provide good 'positive' and 'negative' contrastive examples, and secondly, we fine-tune the base model with a contrastive loss. We describe these two aspects in separate sections below.

4.1 Training Data Augmentation

Contrastive training treats each datapoint as an 'anchor,' and draws the representations of these anchors closer to certain 'positive' representations, and away from 'negative' representations. In the absence of labeled data (i.e., answers to questions), these positives and negatives are constructed by augmenting the data. In this section, we describe the augmentation procedure we use.

Paraphrase Generation. In our approach, positive examples are generated by asking a model to paraphrase the question at hand, and thus we will use the term '*paraphrase*' instead of positive. Importantly, the model used to generate these paraphrases is *the instruct fine tuned version of same as the model being fine-tuned*. This design choice keeps the pipeline self-contained and ensures that the generated paraphrases reflect the linguistic tendencies of the model at hand. It does, however, require an instruction-tuned model. At 7B, generating paraphrases with the base model itself keeps most of the gain (TriviaQA EM 52.08 vs. 58.18; §B.5), and the instruction-tuned model need not come from the same family: SmolLM3-3B trained on Qwen2.5-3B-Instruct generated paraphrases still improves on every benchmark (§B.10).

Templates & Candidate Generation. We use three distinct instruction templates to generate paraphrases. These instructions encourage syntactical variation while preserving the goal of the question: one prompts for a change in grammatical structure, one requests lexical substitutions, and one requests translation into French while preserving English proper nouns. The system prompt instructs the model to return only the rewritten question text with no filler. See §C.5 for details. We generate 5 candidate paraphrases per instruction template to accumulate a total of 15 candidates.

Paraphrase Filtering. A significant challenge with this generation process is that the models frequently produce near-verbatim copies rather than genuine syntactically distinct phrases. We address this by filtering each candidate through a two-stage procedure.

The first stage is a set of heuristic filters, which discard candidates that a) have < 4 words, b) have non-latin characters, c) begin with common filler phrases , d) lack a question mark at the end, or e) share $> 95\%$ word-level overlap with the anchor.

Surviving candidates are evaluated by the base model, which is prompted with few shot examples to answer *Yes* or *No* to whether the candidate asks for the *exact same* factual answer as the anchor. We score candidates using the distribution of the first generated token as $s = p(\textit{Yes})/(p(\textit{Yes}) + p(\textit{No}))$. A candidate paraphrase is retained only if $s_{candidate} \geq 0.80$. Finally, the top four candidates (as per s) are admitted as the paraphrases of the question at hand. The set of paraphrases of a question is denoted $\mathcal{P}$ (with $|\mathcal{P}| = 4$).

Negatives. Small-scale models struggle to generate coherent hard negatives. In pilot experiments, despite prompting for negatives, such models just produced further paraphrases of the anchor. We thus forego model-generated negatives. Instead, we randomly sample 4 questions from the training set (excluding the anchor itself) to generate a set of negative questions $\mathcal{N}$ (with $|\mathcal{N}| = 4$). For diverse training sets of questions (such as TriviaQA), such randomly drawn questions are very unlikely to ask the same question as an anchor, yielding reliable negatives (see §C.5 for details and §B.12 for a near-duplicate audit). Drawing negatives from each anchor's most similar questions instead did not improve accuracy (§B.4).

4.2 Training Objective

The overall loss has two components: a contrastive term that encourages local separation of questions, and a diversity term that limits the global collapse of representations.

Contrastive loss. Let $\mathcal{P}_q$ and $\mathcal{N}_q$ denote the paraphrases and negatives associated with an 'anchor' question q in the training set. We adopt a multi-positive InfoNCE loss formulation (Oord et al., 2018) in which each paraphrase is treated as a positive target, and competes against all remaining paraphrases and negatives. This yields the loss

$$\mathcal{L}_{\text{contrast}} \;=\; -\mathbb{E}_q\left[\frac{1}{|\mathcal{P}_q|}\sum_{p\in\mathcal{P}_q}\log\frac{\exp\big(\text{sim}(q,p)/\alpha\big)}{\sum_{p\in\mathcal{P}_q}\exp\big(\text{sim}(q,p)/\alpha\big)+\sum_{n\in\mathcal{N}_q}\exp\big(\text{sim}(q,n)/\alpha\big)}\right] \tag{3}$$

where $\alpha > 0$ is a temperature hyperparameter. In practice, the expectation is replaced by an empirical average over the training set (or batch, as needed). Including all paraphrases in the denominator follows the standard practice in self-supervised contrastive learning (Chen et al., 2020) and imposes a stricter objective than excluding them: every paraphrase must individually outscore all negatives in the softmax, which implicitly encourages the full set $\mathcal{P}_i$ to cluster tightly around the anchor.

Pairwise diversity loss. The contrastive loss operates within each anchor's neighbourhood and does not prevent different anchors from collapsing to the same point in representation space. We encourage global diversity by penalising the mean pairwise cosine similarity across all anchor representation via

$$\mathcal{L}_{\text{diversity}} \;=\; \mathbb{E}_{q,q'}[\text{sim}(q,q')], \tag{4}$$

where q and q' are drawn independently. In practice, the $\mathbb{E}[\cdot]$ is replaced by an empirical average.

Total objective. The two losses work in tandem: while $\mathcal{L}_{\text{contrast}}$ shapes the local neighbourhood of each anchor, $\mathcal{L}_{\text{diversity}}$ ensures that distinct questions are spread across the representation space globally. With a trade-off parameter $\lambda > 0$, the full training objective is

$$\mathcal{L}_\lambda \;=\; \mathcal{L}_{\text{contrast}} + \lambda\mathcal{L}_{\text{diversity}}. \tag{5}$$

We set $\lambda = 1$ in all experiments. A sweep over λ (§B.3) shows that both terms are needed, and that $\lambda \in [1, 2]$ works best.

5 Experiments

We now evaluate CRAFT's effect on accuracy and on representation collapse.

Model Classes. We apply CRAFT to the base models of the open-source Qwen2.5 family (Qwen et al., 2025) at the 1.5B, 3B, and 7B parameter scales, and to Llama-2-7B (Touvron et al., 2023) to

assess cross-architecture generality. We also apply it to the post-trained SmolLM3-3B (Hugging Face, 2025) (§B.10). All model parameters are updated during training; no layers are frozen.

Training Data. We only use the TriviaQA training split (Joshi et al., 2017) for training both CRAFT and the baselines we study. This training data consists of 10^5 trivia questions drawn from diverse domains. We emphasize again that CRAFT does not use the answers in this split.

Training. We train CRAFT for one epoch over the 10^5 anchor questions with an effective batch size of $B = 16$. We use the AdamW optimiser (Loshchilov & Hutter, 2019) with a flat learning rate of 5×10^{-7} and no learning rate schedule and the temperature is set to $\alpha = 0.07$. All training is performed on one to three NVIDIA RTX 6000 Ada Generation GPUs.

Correctness. Responses use greedy decoding with at most 50 new tokens, and are scored with EM and C-EM (§2).

Evaluation Datasets. We evaluate on the test/validation splits of three open-domain QA benchmarks: **TriviaQA** (Joshi et al., 2017) (single-hop; duplicates removed, ≈10K questions), **NQ-Open** (Kwiatkowski et al., 2019; Lee et al., 2019) (single-hop; ≈3.6K real Google queries answered from memory), and **HotpotQA** (Yang et al., 2018) (multi-hop; ≈7.4K questions).

Since we only train on TriviaQA, the last two measure transfer.

Baselines. We compare against two inference-time methods and two unsupervised fine-tuning methods, all under the same protocol and models (details in §C). **DoLa** (Chuang et al., 2024) contrasts output distributions across layers at decoding time; **ITI** (Li et al., 2024) steers activations along probe-identified truthful directions; **Self-Eval-P(True)** (Zhang et al., 2024) and **FactTune-MC** (Tian et al., 2024) build pseudo-preference data from the model's own confidence and fine-tune with DPO for 3 epochs. CRAFT instead operates only on the hidden states.

Training Compute. DoLA requires no training, and computing the probes for ITI required about $1/2$-GPU-hours. The main training costs were thus associated with CRAFT, FactTune-MC, and Self-Eval-P(True). For Qwen2.5-7B, FactTune-MC required $\approx$ 18GPU-hours/epoch, and Self-Eval $\approx$ 20GPU-hours/epoch. In contrast, CRAFT only takes 12GPU-hours/epoch to execute, thus offering a $> 33\%$ reduction in compute per-epoch. Additionally, we only train CRAFT for one epoch, while FactTune-MC and Self-Eval are trained for 3 epochs each, representing a further $3\times$ speedup. These figures exclude the one-off cost of generating and filtering paraphrases, which in our default pipeline uses the instruction-tuned sibling model (§4).

5.1 CRAFT Improves Accuracy Beyond Baselines

Table 1 reports EM and C-EM accuracy on TriviaQA, NQ-Open, and HotpotQA across three model sizes on the Qwen2.5 family for all baselines, along with the accuracy on the Llama2 architecture for CRAFT, DoLa, and ITI (FactTune-MC, and Self-Eval were omitted due to lack of computational resources, since they take the most training time).

Results on TriviaQA + Qwen. The main observation is a significant increase in accuracy on TriviaQA across model sizes, architectures, and evaluation metrics. The improvements in the EM-accuracy of CRAFT are generally higher than in C-EM. There are two reasons for this softening for C-EM. Firstly, the stronger base model performance under C-EM limits gains, and secondly, part of the effect of CRAFT is reducing the output length (see §A.1), which differentially affects C-EM accuracy.

For the Qwen2.5 family, only CRAFT, FactTune-MC, and Self-Eval consistently give improvements over the base model. For the 3B and 7B models, CRAFT outperforms the best competitor by > 10 points in EM-accuracy, and > 5 points in C-EM-accuracy; at 1.5B the margins are 8.7 and 4.8 points.

Transfer Results for Qwen. The advantage of CRAFT in EM-accuracy persists in this transfer setting: it is the best method on NQ-Open and HotpotQA at every size, 0.5–2.8 points above the next best (FactTune-MC). In C-EM the margins are smaller, but CRAFT improves on the base model in every case, and is best in all but two (a tie at 1.5B and second place at 7B, both on NQ-Open). CRAFT also transfers to WebQuestions (EM $17.08 \rightarrow 28.94$ at 7B; §B.9).

Llama results. On Llama-based model, the C-EM accuracy improvement seen on single-hop benchmarks persists, with a small drop on the multi-hop benchmark HotpotQA. Notably, strict EM accuracy is zero for *all methods* (including base) on Llama, as they consistently continue generating

Table 1: EM and C-EM accuracy (%) on TriviaQA, NQ-Open, and HotpotQA. **Bold**: best per column and model; underline: second best. *Llama-2-7B always answers as `[answer]\n[rand.]` with `rand.` a random string, so standard EM is zero for every method; we compute EM on the first sentence/line. CRAFT uses the all-layer similarity for Llama (§B.2). FactTune-MC and Self-Eval are omitted for Llama for compute reasons.

		TriviaQA		**NQ-Open**		**HotpotQA**	
Model	**Method**	EM	C-EM	EM	C-EM	EM	C-EM
Qwen2.5 1.5B	Base	20.97	34.18	10.80	16.04	11.24	14.41
	DoLa (low)	22.12	31.41	9.89	15.18	11.61	13.98
	DoLa (high)	10.29	21.73	3.38	10.66	6.47	10.05
	ITI	21.22	32.96	10.44	15.71	11.05	13.84
	FactTune-MC	28.02	34.36	12.22	**16.51**	13.14	14.95
	Self-Eval-P(True)	12.50	36.85	7.70	16.37	6.85	13.67
	CRAFT (ours)	**36.72**	**41.61**	**12.69**	**16.51**	**14.91**	**16.31**
Qwen2.5 3B	Base	26.15	39.67	12.91	20.53	12.96	16.95
	DoLa (low)	25.35	33.38	11.83	18.06	12.05	14.36
	DoLa (high)	15.62	23.61	4.82	11.83	6.85	11.49
	ITI	24.39	41.46	10.66	21.00	11.33	16.45
	FactTune-MC	30.80	38.48	15.15	20.44	15.34	17.77
	Self-Eval-P(True)	23.05	43.22	10.75	21.94	11.10	16.38
	CRAFT (ours)	**47.45**	**52.29**	**17.15**	**22.63**	**18.16**	**19.95**
Qwen2.5 7B	Base	38.36	49.81	19.20	26.68	17.85	21.17
	DoLa (low)	32.90	40.23	13.35	20.11	14.58	17.18
	DoLa (high)	25.09	31.74	7.34	12.91	10.45	12.88
	ITI	38.88	49.69	20.03	**27.37**	17.99	21.31
	FactTune-MC	45.63	49.82	22.27	26.68	19.18	20.38
	Self-Eval-P(True)	40.48	53.25	18.01	26.79	18.61	22.19
	CRAFT (ours)	**58.18**	**60.70**	**24.32**	27.20	**21.44**	**22.65**
Llama-2-7b*	Base	49.94	65.60	0.75	26.65	10.93	**21.57**
	DoLa (low)	**52.50**	65.08	9.17	28.64	**11.55**	20.49
	DoLa (high)	51.12	64.89	7.87	28.14	10.70	19.74
	ITI	49.42	65.19	9.67	27.62	10.68	20.88
	CRAFT (ours)	52.33	**66.24**	**12.52**	**29.14**	11.22	21.22

tokens beyond the correct answer. Thus, we report a modified version of EM for Llama. Again, CRAFT produces consistently high accuracy in both criteria, and always either exceeds or nearly-matches competitors. A third family, SmolLM3-3B, shows the same pattern (TriviaQA EM 33.06 $\rightarrow$ 45.61; §B.10), even with paraphrases generated by Qwen2.5-3b-Instruct.

Are the gains just shorter answers? CRAFT makes answers shorter (§A.1), but three controls show this does not explain its gains (§B.8). On the 70% of TriviaQA questions where the 7B base model already answers in 1–2 words, EM still rises from 49.5 to 61.9. Picking the shortest of 10 sampled answers gives shorter outputs than CRAFT (1.78 vs. 1.90 words) but lowers EM to 35.5. Prompting the base model for a short answer lowers EM to 18.7, and raises C-EM only to 53.8 (CRAFT: 60.7).

Other capabilities. On Qwen2.5-7B, CRAFT leaves WikiText-103 perplexity (6.156 vs. 6.159 for the base model), MMLU (74.21 vs. 74.33), and TruthfulQA MC1/MC2 (within 0.6 points) essentially unchanged (§B.9). We do not evaluate instruction following, since CRAFT is applied to base models.

5.2 The Effect of CRAFT on Representations

We now follow up on §3 to investigate the effect of CRAFT on the representation collapse.

CRAFT Eliminates Representation Collapse. Tab. 2 shows summary statistics of the crowdedness scores in the base and CRAFT models on the TriviaQA validation split, at similarity threshold 0.861, the base model's 99th percentile (see §3). CRAFT almost completely eliminates representation collapse at this τ. Histograms of crowdedness are presented in §A.3.

Which neighbours are separated? For each question, we fix the 5% of questions most similar to it under the base model. After CRAFT, the mean similarity to this fixed set falls from 0.813 to 0.561,

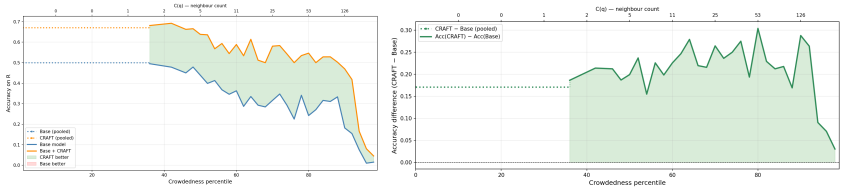


Figure 2: *Left:* Accuracy of the base model and CRAFT against base crowdedness percentiles at similarity 0.861. *Right:* difference of CRAFT and base model accuracy in the same setting. CRAFT increases accuracy at every crowdedness level; the differential grows over most of the range but narrows at the extreme tail.

and it falls for every question. Lexically near-identical pairs with different answers also separate ($0.889 \rightarrow 0.762$ over 100 pairs). However, a question's middle and least similar neighbours move away by comparable amounts, so part of the effect is global. We therefore describe CRAFT as local paraphrase alignment together with global spreading, rather than targeted repulsion of confusable neighbours (§B.6).

Detailed Effect of CRAFT on Accuracy. Fig. 2 plots accuracy against base-model crowdedness percentile (as in Fig. 1) for the base model and for CRAFT at $\tau = 0.861$, together with their difference. This shows how CRAFT's gain is spread over the base model's crowdedness levels.

Table 2: Crowdedness (N_τ^θ) for Qwen2.5-7B at $\tau = 0.861$ over the TriviaQA val. set.

Model	Median	Mean	Max
Base	16	99.6	1246
CRAFT	0	3.8	173

CRAFT improves accuracy across base-crowdedness. Accuracy rises at every base-model crowdedness level, and after CRAFT it is above the overall base-model accuracy at every level. Even a few nearby questions may cause confusion, consistent with the smooth decline in Fig. 1.

CRAFT's gain varies with crowdedness but stays positive. At 7B, the gain rises from about $+0.13$ at the least crowded resolved bin to about $+0.28$ in the upper-middle range, and falls to about $+0.05$ at the most crowded percentiles. At 1.5B and 3B, crowded and uncrowded questions gain by similar amounts, with the most crowded questions gaining slightly less (§A.2).

Since CRAFT uses no answers, its gains are unlikely to come from new knowledge, and more likely from better retrieval of knowledge already in the weights. Together with the large drop in crowdedness, this is consistent with representation collapse hurting retrieval. This is not a causal study: we neither identify nor control for potential confounders.

6 Discussion

We showed that representation collapse is strongly associated with failures of factual retrieval, and proposed CRAFT, an unsupervised contrastive method that reshapes the hidden-state geometry instead of adding information or building pseudo-labels. CRAFT improves accuracy on single- and multi-hop benchmarks across sizes and architectures, beats prior unsupervised fine-tuning and inference-time methods at lower training cost, and leaves general capabilities essentially unchanged. Its gains coincide with the decongestion it produces, and appear at every level of crowdedness.

Limitations. The link between collapse and accuracy is associative, not causal: CRAFT does not control for confounders, and entity frequency, knowledge availability, and question form remain plausible common causes. Identifying and controlling for them is an important direction for future work. CRAFT also acts partly through global spreading rather than targeted separation of confusable questions (§5.2). Finally, our best results use paraphrases from an instruction-tuned sibling of the base model, and we have not evaluated instruction following after CRAFT.

AI use statement

We used generative AI tools to help write code for the experiments and figures, and to edit the writing. We did not use them for any other research subtask. The authors reviewed all AI-assisted code and text and take full responsibility for the content of this paper.

REPRODUCIBILITY STATEMENT

§4 and §C.5 describe the full data pipeline, prompts, filters, and training settings, and §C gives the settings of every baseline.

REFERENCES

Jonathan Berant, Andrew Chou, Roy Frostig, and Percy Liang. Semantic parsing on Freebase from question-answer pairs. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pp. 1533–1544, 2013.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 1597–1607. PMLR, 2020. URL https://proceedings.mlr.press/v119/chen20j.html.

Jiahao Cheng, Tiancheng Su, Jia Yuan, Guoxiu He, Jiawei Liu, Xinqi Tao, Jingwen Xie, and Huaxia Li. Chain-of-thought prompting obscures hallucination cues in large language models: An empirical evaluation, 2025. URL https://arxiv.org/abs/2506.17088.

Yung-Sung Chuang, Yujia Xie, Hongyin Luo, Yoon Kim, James R. Glass, and Pengcheng He. Dola: Decoding by contrasting layers improves factuality in large language models. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=Th6NyL07na.

Zorik Gekhman, Eyal Ben David, Hadas Orgad, Eran Ofek, Yonatan Belinkov, Idan Szpektor, Jonathan Herzig, and Roi Reichart. Inside-out: Hidden factual knowledge in llms, 2025. URL https://arxiv.org/abs/2503.15299.

Aryo Pradipta Gema, Chen Jin, Ahmed Abdulaal, Tom Diethe, Philip Teare, Beatrice Alex, Pasquale Minervini, and Amrutha Saseendran. Decore: Decoding by contrasting retrieval heads to mitigate hallucinations, 2024. URL https://arxiv.org/abs/2410.18860.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. Measuring massive multitask language understanding, 2021. URL https://arxiv.org/abs/2009.03300.

Hugging Face. SmolLM3: smol, multilingual, long-context reasoner. https://huggingface.co/blog/smollm3, 2025.

Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Yejin Bang, Delong Chen, Wenliang Dai, Ho Shu Chan, Andrea Madotto, and Pascale Fung. Survey of hallucination in natural language generation. *ACM Computing Surveys*, 55(12):1–38, 2023. doi: 10.1145/3571730. URL https://doi.org/10.1145/3571730.

Bowen Jin, Hansi Zeng, Zhenrui Yue, Jinsung Yoon, Sercan Arik, Dong Wang, Hamed Zamani, and Jiawei Han. Search-r1: Training llms to reason and leverage search engines with reinforcement learning, 2025. URL https://arxiv.org/abs/2503.09516.

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

Saurav Kadavath, Tom Conerly, Amanda Askell, Tom Henighan, Dawn Drain, Ethan Perez, Nicholas Schiefer, Zac Hatfield-Dodds, Nova DasSarma, Eli Tran-Johnson, Scott Johnston, Sheer El-Showk, Andy Jones, Nelson Elhage, Tristan Hume, Anna Chen, Yuntao Bai, Sam Bowman, Stanislav Fort, Deep Ganguli, Danny Hernandez, Josh Jacobson, Jackson Kernion, Shauna Kravec, Liane Lovitt, Kamal Ndousse, Catherine Olsson, Sam Ringer, Dario Amodei, Tom Brown, Jack Clark, Nicholas Joseph, Ben Mann, Sam McCandlish, Chris Olah, and Jared Kaplan. Language models (mostly) know what they know, 2022. URL https://arxiv.org/abs/2207.05221.

Adam Tauman Kalai, Ofir Nachum, Santosh S. Vempala, and Edwin Zhang. Why language models hallucinate, 2025. URL https://arxiv.org/abs/2509.04664.

Nikhil Kandpal, Haikang Deng, Adam Roberts, Eric Wallace, and Colin Raffel. Large language models struggle to learn long-tail knowledge, 2023. URL https://arxiv.org/abs/2211.08411.

Guy Kaplan, Zorik Gekhman, Zhen Zhu, Lotem Rozner, Yuval Reif, Swabha Swayamdipta, Derek Hoiem, and Roy Schwartz. Why fine-tuning encourages hallucinations and how to fix it, 2026. URL https://arxiv.org/abs/2604.15574.

Aayush Karan and Yilun Du. Reasoning with sampling: Your base model is smarter than you think, 2025. URL https://arxiv.org/abs/2510.14901.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, et al. Natural questions: a benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:453–466, 2019.

Kenton Lee, Ming-Wei Chang, and Kristina Toutanova. Latent retrieval for weakly supervised open domain question answering. In *Proceedings of the 57th annual meeting of the association for computational linguistics*, pp. 6086–6096, 2019.

Kenneth Li, Oam Patel, Fernanda Viégas, Hanspeter Pfister, and Martin Wattenberg. Inference-time intervention: Eliciting truthful answers from a language model, 2024. URL https://arxiv.org/abs/2306.03341.

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

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019. URL https://openreview.net/forum?id=Bkg6RiCqY7.

Alex Mallen, Akari Asai, Victor Zhong, Rajarshi Das, Daniel Khashabi, and Hannaneh Hajishirzi. 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)*, pp. 9802–9822. Association for Computational Linguistics, 2023. doi: 10.18653/v1/2023.acl-long.546. URL https://aclanthology.org/2023.acl-long.546/.

Kevin Meng, David Bau, Alex Andonian, and Yonatan Belinkov. Locating and editing factual associations in GPT. In *Advances in Neural Information Processing Systems*, volume 35, pp. 17359–17372, 2022. URL https://proceedings.neurips.cc/paper_files/paper/2022/hash/6f1d43d5a82a37e89b0665b33bf3a182-Abstract-Conference.html.

Kevin Meng, Arnab Sen Sharma, Alex Andonian, Yonatan Belinkov, and David Bau. Mass-editing memory in a transformer. In *International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=MkbcAHIYgyS.

Stephen Merity, Caiming Xiong, James Bradbury, and Richard Socher. Pointer sentinel mixture models, 2016. URL https://arxiv.org/abs/1609.07843.

Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.

Sangjun Park, Elliot Meyerson, Xin Qiu, and Risto Miikkulainen. Fine-tuning language models to know what they know, 2026. URL https://arxiv.org/abs/2602.02605.

Qwen, :, An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, Keming Lu, Keqin Bao, Kexin Yang, Le Yu, Mei Li, Mingfeng Xue, Pei Zhang, Qin Zhu, Rui Men, Runji Lin, Tianhao Li, Tianyi Tang, Tingyu Xia, Xingzhang Ren, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yu Wan, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zihan Qiu. Qwen2.5 technical report, 2025. URL https://arxiv.org/abs/2412.15115.

Katherine Tian, Eric Mitchell, Huaxiu Yao, Christopher D Manning, and Chelsea Finn. Fine-tuning language models for factuality. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=WPZ2yPag4K.

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

Cunxiang Wang, Xiaoze Liu, Yuanhao Yue, Xiangru Tang, Tianhang Zhang, Cheng Jiayang, Yunzhi Yao, Wenyang Gao, Xuming Hu, Zehan Qi, et al. Survey on factuality in large language models: Knowledge, retrieval and domain-specificity. *arXiv preprint arXiv:2310.07521*, 2023.

Boyang Xue, Fei Mi, Qi Zhu, Hongru Wang, Rui Wang, Sheng Wang, Erxin Yu, Xuming Hu, and Kam-Fai Wong. Ualign: Leveraging uncertainty estimations for factuality alignment on large language models, 2025. URL https://arxiv.org/abs/2412.11803.

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

Xiaoying Zhang, Baolin Peng, Ye Tian, Jingyan Zhou, Lifeng Jin, Linfeng Song, Haitao Mi, and Helen Meng. Self-alignment for factuality: Mitigating hallucinations in llms via self-evaluation, 2024. URL https://arxiv.org/abs/2402.09267.

A REPRESENTATIONAL COLLAPSE AND CRAFT

We study further aspects of representation collapse by characterising how the behaviour of the base model varies with crowdedness, and also highlight the effects of CRAFT in more details.

A.1 CROWDEDNESS VERSUS ANSWER LENGTH, AND THE EFFECT OF CRAFT.

Answer lengths and Crowdedness. While §3 and §5.2 focus on the relationship between crowdedness and accuracy, it turns out that similar relationship holds between crowdedness, and the length of the model response. Fig. 3 demonstrates this effect in the same setup as Fig. 1 from §3. Observe that answers get much longer as crowdedness increases: from about 3 words at low crowdedness to about 7.5–10 words at the most crowded end. This indicates that as questions get more crowded, the corresponding responses get longer, and less precise. We suspect that this is a signature of local ambiguity, in that when the model cannot separate the question from its dense neighbourhood, the distribution of the response flattens, and leads to more verbose outputs. This is illustrated qualitatively in Table 4, where the base (and many competitors) produce verbose responses which give multiple hedging answers and irrelevant information to a single pointed question.

Note that the effect of answer length on accuracy is nontrivial. For instance, in the stringent EM-sense, a very long answer is likely to be marked incorrect for generating too much extra text. On the other hand, in the more relaxed C-EM sense, this extra text gives the model a better chance to produce the correct answer within its output.

CRAFT Reduces Answer Lengths. As also highlighted in Fig. 3 (and Figure 4 for Llama-2-7B model), CRAFT has a stark effect on answer lengths, and makes the model significantly more concise across base-crowdedness levels. Table 3 gives mean values of these answer lengths, again demonstrating more concise outputs. The same can again be seen qualitatively in Table 4.

Let us emphasize that although CRAFT causes a reduction in answer length, it's improvement in accuracy cannot be simply attributed to this. While certainly this reduction can improve the EM-accuracy, such a reduction actually hurts C-EM-accuracy, and so CRAFT's gains in this metric (as shown in §5) cannot be explained away by such conciseness. In fact, CRAFT also improves the accuracy of questions whose base answers are already short, and simply making answers shorter does not reproduce its gains (§B.8). Table 5 gives a qualitative illustration of this effect, where all competing models give incorrect 2 or 3 word answer, but craft finds the correct 2 word answer.

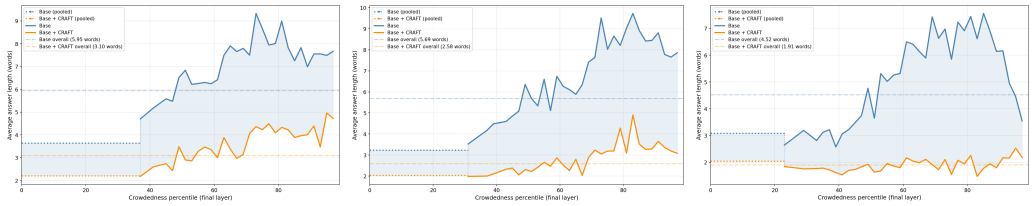


Figure 3: Average answer length of TriviaQA questions vs. crowdedness percentile for Qwen2.5-1.5B (*left*), 3B (*center*), and 7B (*right*) at the 99th percentile similarity threshold. The base model becomes more verbose as crowdedness grows, peaking at about 9–10 words (1.5B, 3B) and 7.5 words (7B). CRAFT answers are shorter at every crowdedness level: about 2 words throughout at 7B, and 2–5 words at 1.5B and 3B.

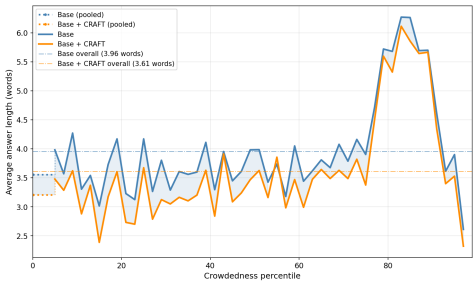


Figure 4: Average answer length vs. crowdedness percentile for Llama-2-7B at the 99th percentile similarity threshold on TriviaQA. Unlike Qwen2.5-based models, Llama-2-7B exhibits a less pronounced verbosity suppression effect after CRAFT fine-tuning, though the effect persists: CRAFT consistently produces shorter outputs across all crowdedness levels.

Table 3: Average response length (in words) for the first generated answer sentence of the base model and CRAFT across all benchmarks and model sizes.

Model	Method	TriviaQA	NQ-Open	HotpotQA
Qwen2.5-1.5B	Base	5.95	7.39	7.36
	CRAFT	3.10	5.38	3.81
Qwen2.5-3B	Base	5.69	7.39	6.92
	CRAFT	2.58	5.04	3.19
Qwen2.5-7B	Base	4.52	6.87	6.51
	CRAFT	1.91	3.46	2.50
Llama-2-7B	Base	3.94	17.23	6.23
	CRAFT	3.61	8.27	5.57

Table 4: Qualitative comparison of Qwen2-5 7B base and CRAFT fine tuned model outputs on a representative TriviaQA question.

Question	What did Antonio Stradivari make?
Correct answer	Violins
Base model	Antonio Stradivari, a famous Italian luthier, made violins, violas, cellos, and other stringed instruments. His violins are known for their beautiful sound and are highly valued by musicians and collectors.
Dola (Low)	In 1720, the most famous luthier in history, Antonio Stradivari, moved his workshop to via del Pigna (now via Guglielmo Marconi) and started building some of his largest and ...
Dola (High)	The 2013 film, Ḧildegard of Bingen̈(in German and French, with English subtitles) is a dramatized look at the life and work of the 12th century composer, writer, and ...
ITI	Antonio Stradivari, a famous Italian luthier, made some of the most renowned stringed instruments in history, including violins, violas, cellos, and double basses. His instruments are highly sought after for their exceptional craftsmanship ...
FactTune-MC	Antonio Stradivari was a famous Italian luthier who made some of the most beautiful and valuable stringed instruments in history.
Self-Eval-P(True)	Antonio Stradivari, a famous Italian luthier, made some of the most renowned stringed instruments in history, including violins, violas, cellos, and double basses.
CRAFT	Violin

Table 5: Qualitative comparison of Qwen2-5 3B base and CRAFT fine tuned model outputs on a representative TriviaQA question.

Question	Beloved in 1999 was whose first movie since The Color Purple in 1985?
Correct answer	Oprah Winfrey
Base model	cary grant
Dola (Low)	cecil b demille
Dola (High)	gabrielle andrew
ITI	cecil b. demille
FactTune-MC	cecil b demille
Self-Eval-P(True)	cecil b. demille
CRAFT	Oprah Winfrey

A.2 Another look at the effect of Representational Collapse on Accuracy, and on CRAFT's differential improvement.

Accuracy heatmaps. To express the overall, rather than point-wise with crowdedness behaviour of the accuracy improvement via CRAFT, we study the difference of accuracy in more-crowded versus less-crowded questions. More concretely, for $\pi \in (0, 100)$, let $\mathcal{R}_\tau(\pi)$ denote the $\pi\%$ of questions with the highest base-model crowdedness $N_\tau^{\text{BASE}}(q)$. In this section, we are concerned with the differential

$$\delta_{\tau,\pi} := \text{Acc}((\mathcal{R}_\tau(\pi))^c) - \text{Acc}(\mathcal{R}_\tau(\pi)).$$

This integrates to the left and right of a point in the curves of Fig. 1, and considers the difference.

Fig. 5 shows $\delta_{\tau,\pi}$ as π and τ vary. The gap is positive almost everywhere: crowded questions are answered correctly less often across nearly all choices of τ and π. At 1.5B and 3B it is largest for τ between the 80th and 95th percentiles; at 7B it grows with τ and nearly disappears at the lowest thresholds, consistent with Fig. 1.

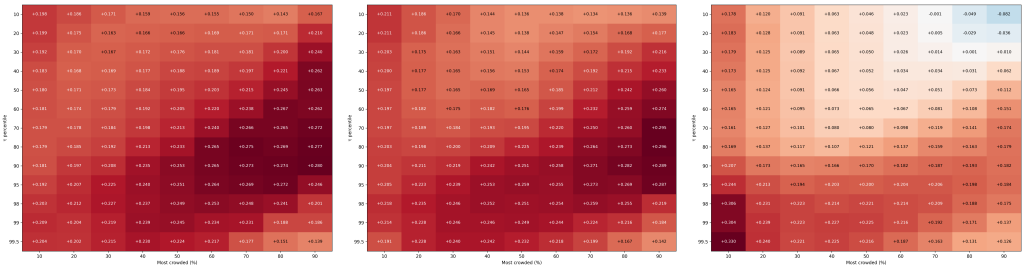
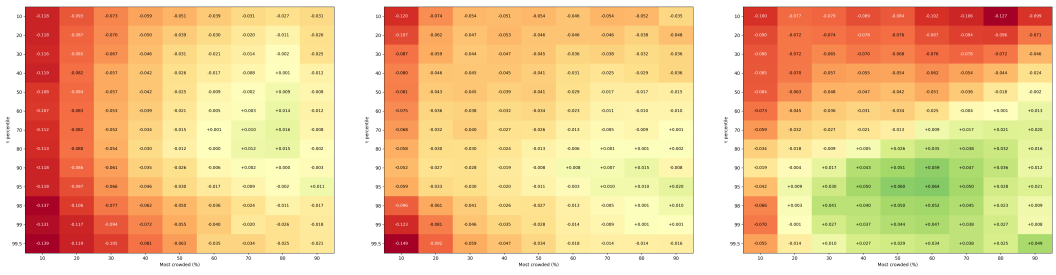


Figure 5: Base-model accuracy gap $\delta_{\tau,\pi} = \text{Acc}(\mathcal{R}^c) - \text{Acc}(\mathcal{R})$ on TriviaQA (EM) for Qwen2.5-1.5B (left), 3B (middle), and 7B (right). $\mathcal{R}$ is the most crowded $\pi\%$ of questions (x-axis), with crowdedness counted at threshold τ, given as a percentile of all pairwise final-layer similarities (y-axis). Red cells mean crowded questions are answered correctly less often; blue cells mean more often. The gap is positive in almost every cell, reaching about 33 points. For 1.5B and 3B it is largest at τ between the 80th and 95th percentiles. For 7B it grows with τ and is largest at the strictest thresholds, especially for the most crowded 10%.

Effect of CRAFT. We also ask whether CRAFT's gains concentrate on crowded questions, using

$$\Delta_{\tau,\pi} := \left[\text{Acc}_{\text{CRAFT}}(\mathcal{R}) - \text{Acc}_{\text{BASE}}(\mathcal{R})\right] - \left[\text{Acc}_{\text{CRAFT}}(\mathcal{R}^c) - \text{Acc}_{\text{BASE}}(\mathcal{R}^c)\right].$$

Positive values mean crowded questions gain more than the rest. Fig. 6 shows that $\Delta_{\tau,\pi}$ is small across most of the grid: crowded and uncrowded questions gain by similar amounts. At 7B, moderately crowded questions gain slightly more; at 1.5B and 3B, the most crowded questions gain slightly less.

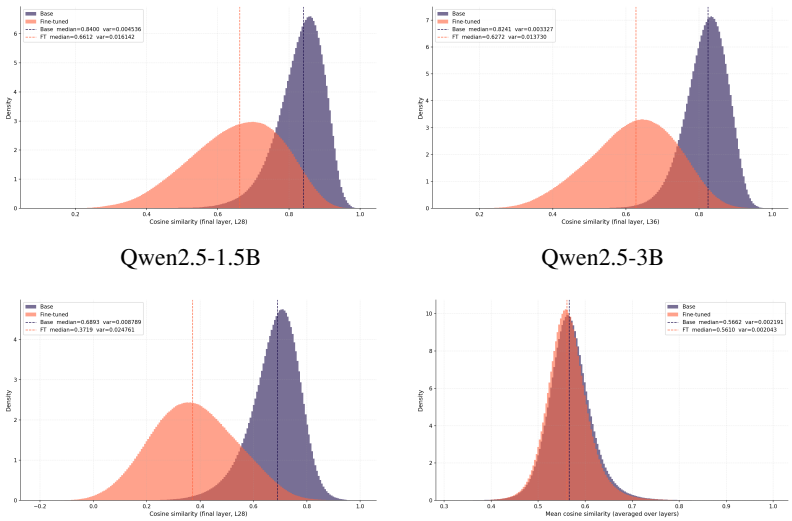


(a) 1.5B (b) 3B (c) 7B

Figure 6: CRAFT improvement differential $\Delta_{\tau,\pi} = [\mathrm{Acc}_{\mathrm{CRAFT}}(\mathcal{R}) - \mathrm{Acc}_{\mathrm{BASE}}(\mathcal{R})] - [\mathrm{Acc}_{\mathrm{CRAFT}}(\mathcal{R}^c) - \mathrm{Acc}_{\mathrm{BASE}}(\mathcal{R}^c)]$ on TriviaQA (EM) for Qwen2.5-1.5B (left), 3B (middle), and 7B (right). $\mathcal{R}$ is the most crowded $\pi\%$ of questions under the base model (x-axis), with crowdedness counted at threshold τ, given as a percentile of all pairwise final-layer similarities (y-axis). Green cells mean CRAFT improves crowded questions more than the rest; red cells mean less. For 1.5B and 3B, the most crowded questions gain less than the rest, and the differential is close to zero elsewhere. For 7B, at high τ, CRAFT favours moderately crowded questions (up to about $+0.06$), while the most crowded 10% gain slightly less.

A.3 Hidden State Similarity Across Questions, and Crowdedness histograms

We measure the cosine similarity of question representations (final layer for Qwen2.5, all-layer average for Llama-2-7B) across TriviaQA questions to assess how fine-tuning affects the geometry of internal representations. As shown in Figure 7, CRAFT lowers the median cosine similarity between question representations at every Qwen2.5 size, so representations become more dispersed; for Llama-2-7B, the shift is much smaller. For the 1.5B model, the median similarity drops from 0.840 to 0.661; for the 3B model, from 0.824 to 0.627; and for the Qwen-7B model, from 0.689 to 0.372. This suggests that fine-tuning encourages the model to form more distinct internal representations for different inputs, rather than mapping questions to a narrow cluster in representation space.



Qwen2.5-1.5B Qwen2.5-3B

Qwen2.5-7B Llama-2 7B

Figure 7: Distributions of pairwise cosine similarity between question representations (final layer for Qwen2.5, all-layer average for Llama-2-7B) over TriviaQA question pairs, before and after fine-tuning. Dashed lines indicate the median of each distribution.

This decrowding effect is also visible in the crowdedness histograms (Fig. 8) per threshold accross all models and all benchmark. Indeed the CRAFT tuning manages to separate the hidden state representations of different questions from one another.

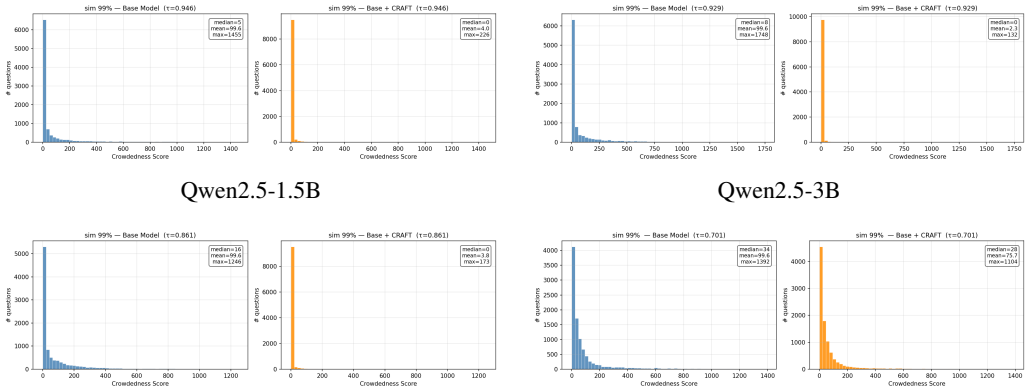
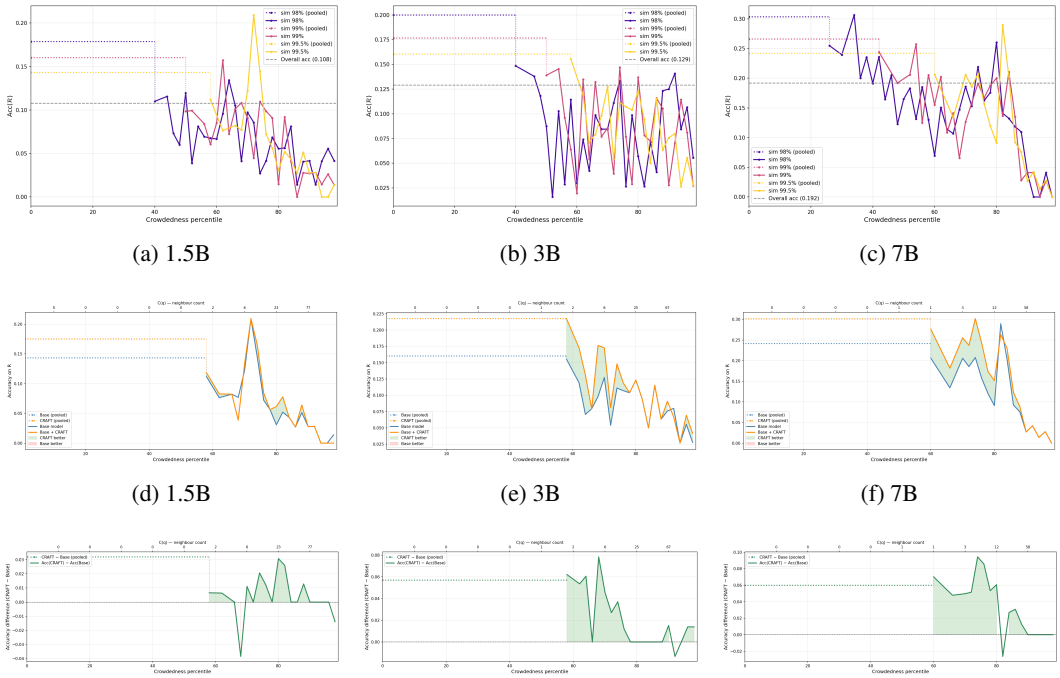


Figure 8: Distributions of crowdedness (final layer for Qwen2.5, all-layer average for Llama-2-7B) of questions at 99-percentile similarity levels for Qwen2.5 models and Llama-2 7B over the TriviaQA validation set. The Base model (left) sees a large degree of crowdedness, which CRAFT almost entirely eliminates (right).

A.4 CROWDEDNESS VS ACCURACY ON NQ-OPEN AND HOTPOTQA, AND FOR LLAMA

Crowdedness vs Accuracy on NQ-Open: Figure 9 shows accuracy versus crowdedness percentile on NQ-Open for all three model sizes. As on TriviaQA, base-model accuracy falls as crowdedness rises, at every size and threshold, and is close to zero for the most crowded questions. CRAFT improves accuracy over most of the crowdedness range, but like on TriviaQA, its gains are not concentrated on the most crowded questions: they are largest in the middle of the range and near zero in the top few percentiles, where both models are rarely correct.



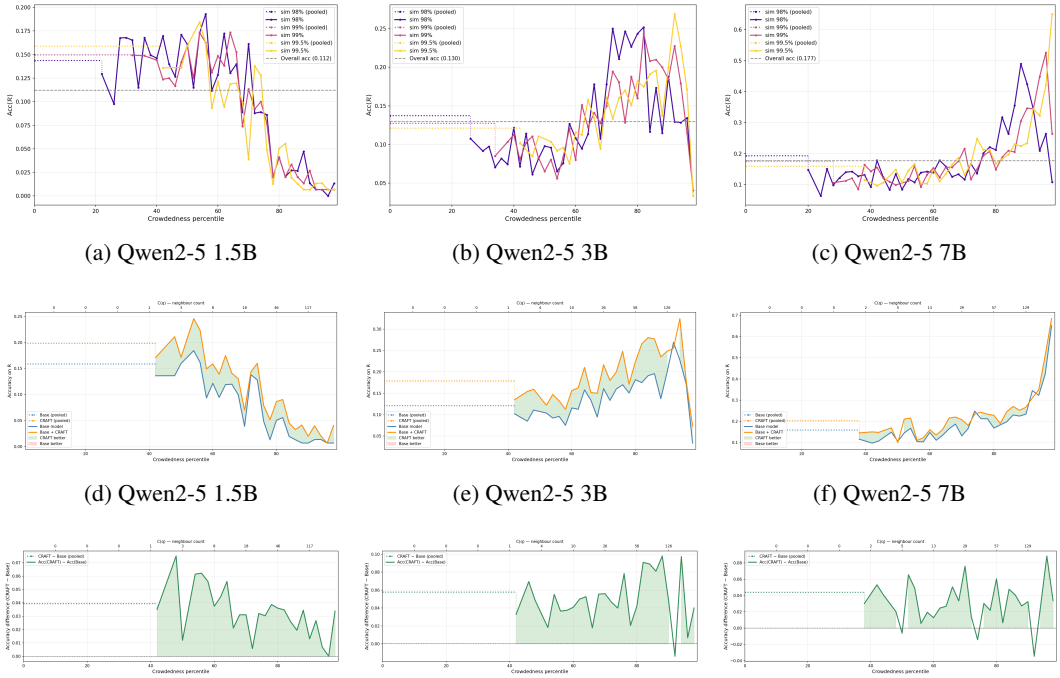
(a) 1.5B (b) 3B (c) 7B

(d) 1.5B (e) 3B (f) 7B

(g) 1.5B (h) 3B (i) 7B

Figure 9: NQ-Open results across model sizes (1.5B, 3B, 7B). *Top row:* Accuracy on crowded questions as a function of crowdedness percentile, shown separately for three similarity thresholds (98%, 99%, 99.5%). Shaded regions indicate percentile bins with no data. The dashed horizontal line denotes overall accuracy. *Middle row:* Accuracy as a function of crowdedness percentile for 99th percentile similarity score for the base model and the CRAFT fine tuned base model, with shading indicating regions where one model outperforms the other. *Bottom row:* Per-bin accuracy difference between the fine-tuned and base models as a function of crowdedness for 99th percentile similarity score.

Crowdedness vs Accuracy on Hotpot-QA: Figure 10 shows accuracy versus crowdedness percentile on HotpotQA for all three model sizes. Here the picture differs from the single-hop benchmarks. For 1.5B, base-model accuracy falls with crowdedness as before, but for 3B and 7B it rises with crowdedness over most of the range. One possible explanation is that HotpotQA is multi-hop: questions that share a template may also share a reasoning pattern the model handles well. CRAFT improves accuracy across most of the crowdedness range at every size, without a clear concentration on the most crowded questions.



(a) Qwen2-5 1.5B (b) Qwen2-5 3B (c) Qwen2-5 7B

(d) Qwen2-5 1.5B (e) Qwen2-5 3B (f) Qwen2-5 7B

(g) Qwen2-5 1.5B (h) Qwen2-5 3B (i) Qwen2-5 7B

Figure 10: HotpotQA results across model sizes (1.5B, 3B, 7B). *Top row:* Accuracy on crowded questions as a function of crowdedness percentile, shown separately for three similarity thresholds (98%, 99%, 99.5%). Shaded regions indicate percentile bins with no data. The dashed horizontal line denotes overall accuracy. *Middle row:* Accuracy as a function of crowdedness percentile for 99th percentile similarity score for the base model and the CRAFT fine tuned base model, with shading indicating regions where one model outperforms the other. *Bottom row:* Per-bin accuracy difference between the fine-tuned and base models as a function of crowdedness for 99th percentile similarity score.

Crowdedness v/s Accuracy for Llama2-7B. Finally, Figure 11 presents the crowdedness–accuracy relationship for Llama2-7B across all three evaluation benchmarks, comparing the base model against its CRAFT fine-tuned counterpart. The top row shows accuracy on the crowded set $\mathcal{R}$ as a function of crowdedness percentile across multiple similarity thresholds τ, the middle row overlays base and fine-tuned accuracy curves with shading indicating where each model has the advantage, and the bottom row shows the pointwise accuracy difference $\mathrm{Acc}(\mathrm{CRAFT}) - \mathrm{Acc}(\mathrm{Base})$ as a function of crowdedness.

The pattern varies across datasets: on TriviaQA, accuracy decreases gradually as crowdedness increases before spiking at the extreme tail. On NQ-Open, the accuracy drops sharply and consistently as crowdedness rises. HotpotQA shows the opposite trend, where accuracy increases with crowdedness, mirroring the behaviour observed in Qwen2.5-based models and suggesting that highly crowded multi-hop questions may share retrievable reasoning patterns that the model exploits.

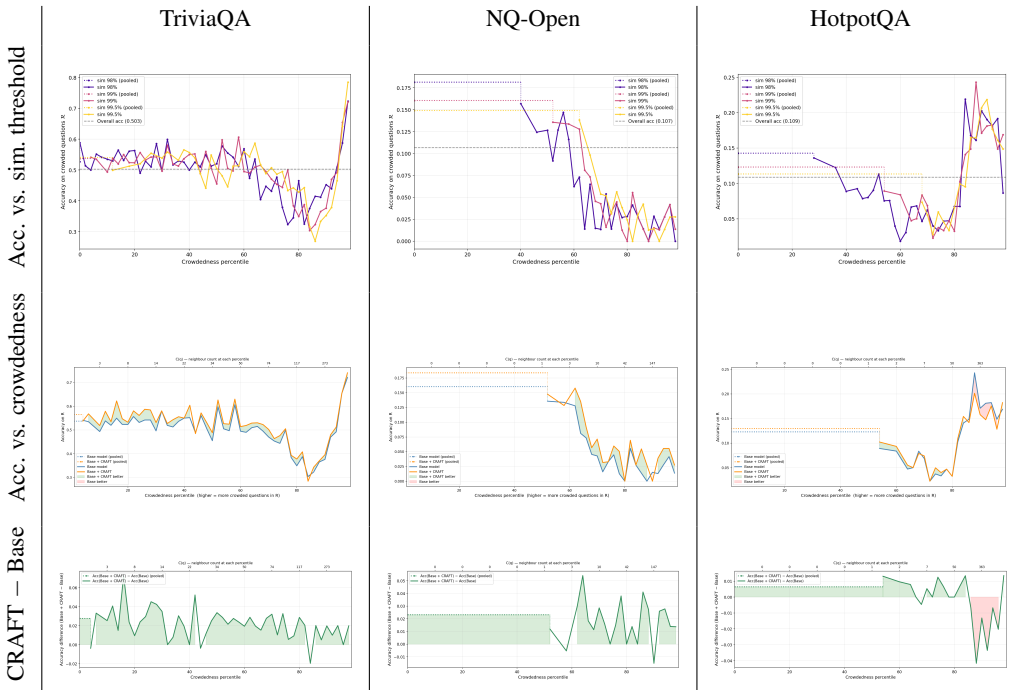


Figure 11: Crowdedness-accuracy analysis for Llama-2 7B across TriviaQA, NQ-Open, and HotpotQA. On TriviaQA, accuracy initially decreases in more crowded regions before jumping at the extreme crowdedness tail. On NQ-Open, accuracy drops sharply as crowdedness increases. On HotpotQA, accuracy improves substantially on the most crowded questions.

B Ablations, Controls, and Additional Evaluations

Unless stated otherwise, all results in this section use Qwen2.5-7B, final-layer similarity, the TriviaQA validation set, and the default CRAFT settings of §5.

B.1 Layer Study

Which layer carries the signal? For each layer ℓ, we compute crowdedness using only the similarity at that layer, with τ set to the 99th percentile of that layer's similarity scores. We then measure the correlation ρ of crowdedness with accuracy, and with the entropy of the first generated answer token. The entropy needs no answers, so it gives an answer-free view of the same association. Table 6 shows that both correlations are near zero in early and middle layers, grow sharply from layer 24 on, and are strongest at the final layer. Here ρ is the Spearman correlation, computed over all TriviaQA validation questions, and τ is the 99th percentile of that layer's pairwise similarities.

Table 6: Correlation of per-layer crowdedness with accuracy and with first-token entropy, Qwen2.5-7B (28 layers).

Layer	1	4	8	12	16	18	20	22	24	25	26	27	28
τ	.995	.990	.979	.978	.968	.957	.910	.860	.715	.679	.666	.693	.861
ρ (accuracy)	$+.013$	$-.041$	$+.018$	$+.015$	$+.013$	$-.049$	$-.040$	$-.067$	$-.176$	$-.179$	$-.188$	$-.179$	$\mathbf{-.253}$
ρ (entropy)	$-.039$	$+.066$	$-.010$	$-.013$	$-.036$	$+.004$	$+.037$	$+.088$	$+.147$	$+.123$	$+.127$	$+.108$	$\mathbf{+.311}$

Which similarity should CRAFT use? We trained CRAFT with four versions of the similarity in Eq. (2): all layers weighted equally (§B.2), layers 24 onward weighted equally, layers 24 onward weighted by their accuracy correlation, and the final layer only. Table 7 shows that the final layer alone works best. Restricting to layers 24 onward is worse than using all layers on TriviaQA, with or without weights. We therefore use the final layer for all Qwen2.5 results.

Table 7: CRAFT trained with different layer choices for the similarity, Qwen2.5-7B.

	TriviaQA		NQ-Open		HotpotQA	
Layers used	EM	C-EM	EM	C-EM	EM	C-EM
All layers	55.52	58.68	23.38	26.70	20.16	21.53
Layers 24–28	48.46	52.14	23.27	26.62	20.51	22.04
Layers 24–28, weighted	49.27	52.82	23.24	26.68	20.58	22.04
Final layer only (default)	**58.18**	**60.70**	**24.32**	**27.20**	**21.44**	**22.65**

Final-layer similarity also helps at the other sizes: compared with all layers, TriviaQA EM rises from 33.47 to 36.72 at 1.5B and from 44.46 to 47.45 at 3B (Table 1 reports the final-layer results).

B.2 All-layer similarity

For Llama-2-7B, we measure the similarity between two questions using the hidden states of all layers, instead of the final layer alone. We stack the hidden states of the last prompt token across all layers into a matrix:

$$\mathbf{H}_\theta(q) = \left[h_0^\theta(q);\, h_1^\theta(q);\, \ldots;\, h_L^\theta(q)\right] \in \mathbb{R}^{(L+1)\times d}. \tag{6}$$

This *stacked* representation treats each layer's contribution equally, rather than focusing on any one layer, and so allows for signals spread across layers. Fine-tuning on this representation also lets the contrastive signal act on all layers directly. The similarity between two questions q and q' is the mean of the per-layer cosine similarities:

$$\mathrm{sim}_\theta^{\mathrm{all}}(q, q') = \frac{1}{L+1}\sum_{\ell=0}^{L} \frac{h_\ell^\theta(q)^\top h_\ell^\theta(q')}{\|h_\ell^\theta(q)\|\,\|h_\ell^\theta(q')\|}. \tag{7}$$

For Llama-2-7B, this replaces Eq. (2) everywhere: in the crowdedness score and in both CRAFT losses (Eqs. 3–4). It is also the "all layers" variant in Table 7.

Why Llama-2-7B uses all layers. Training CRAFT on Llama-2-7B with final-layer similarity lowered TriviaQA EM below the base model (43.18 vs. 49.94), while the all-layer variant raised it to 52.33. Table 8 suggests why. With final-layer similarity, CRAFT does not spread Llama-2-7B's representations: their median final-layer similarity is unchanged, and their all-layer similarity slightly increases. With all-layer similarity, CRAFT lowers similarity under both measures, including at the final layer. Llama-2-7B is also much less crowded than Qwen2.5 to begin with (median final-layer similarity 0.45, vs. 0.69–0.84 for Qwen2.5; Figure 12 and §A.3), and CRAFT changes its geometry much less than it does for Qwen2.5. With less crowding to remove, this is a likely reason for CRAFT's smaller gains on Llama-2-7B (Table 1). As elsewhere in the paper, this is only an association: we cannot isolate the effect of crowdedness from other differences between the two model families, and we do not read it as causal.

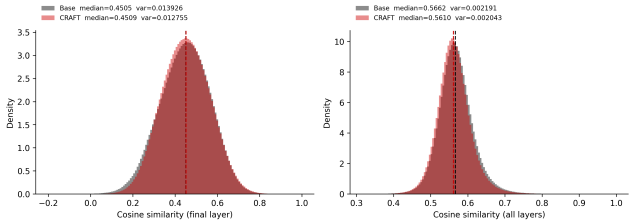


Figure 12: Distribution of pairwise cosine similarity between TriviaQA question representations for base Llama-2-7B, using the final layer only and the average over all layers. Dashed lines mark the medians.

Table 8: Median pairwise cosine similarity between TriviaQA question representations for Llama-2-7B, under final-layer and all-layer similarity.

Model	Final layer	All layers
Base	0.451	0.566
CRAFT, trained with final-layer similarity	0.451	0.586
CRAFT, trained with all-layer similarity	0.416	0.561

B.3 Loss Ablation

Table 9 varies λ in Eq. (5), including each term on its own. Both ends do worse than the full objective, and $\lambda = 1$ and $\lambda = 2$ are close. The contrastive term cannot be tested with its paraphrases removed, since InfoNCE then has no positive target. So this table isolates the contrastive term as a whole, and §B.4 isolates the choice of negatives.

Table 9: Effect of the trade-off λ between the contrastive and diversity terms, Qwen2.5-7B.

	TriviaQA		NQ-Open		HotpotQA	
Objective	EM	C-EM	EM	C-EM	EM	C-EM
Contrastive only ($\lambda = 0$)	46.21	49.65	22.60	26.29	18.73	20.27
$\lambda = 0.5$	48.34	50.96	23.60	26.65	20.01	21.42
$\lambda = 1$ (default)	**58.18**	**60.70**	24.32	**27.20**	**21.44**	**22.65**
$\lambda = 2$	57.80	60.09	**24.49**	**27.20**	21.42	22.47
Diversity only	53.49	57.63	23.21	27.04	20.51	22.34

B.4 Choice of Negatives

Hard negatives usually give a stronger contrastive signal, but small models cannot generate them reliably (§4). As a simple alternative, we chose four negatives at random from each anchor's 20 most similar training questions. Table 10 shows that this does not help, and slightly lowers accuracy.

Table 10: Random vs. similarity-selected negatives, Qwen2.5-7B.

	TriviaQA		NQ-Open		HotpotQA	
Negatives	EM	C-EM	EM	C-EM	EM	C-EM
Similarity-selected	56.12	59.91	24.18	27.01	21.41	22.62
Random (default)	**58.18**	**60.70**	**24.32**	**27.20**	**21.44**	**22.65**

B.5 Paraphrase Source

Our default pipeline generates paraphrases with the instruction-tuned sibling of the base model. We test two alternatives. First, we generated and filtered paraphrases with the base Qwen2.5-7B model alone (Table 11). This keeps almost all of the C-EM gain and most of the TriviaQA EM gain (52.08 vs. 58.18), but on NQ-Open and HotpotQA, EM falls to or below the base model. The instruction-tuned paraphrases therefore seem to matter mostly for answer format, and our best EM results depend on them. Second, the instruction-tuned model need not come from the same family: SmolLM3-3B trained on paraphrases from the Qwen2.5-3B pipeline improves on every benchmark (§B.10). Using an instruction-tuned model adds a preprocessing step, but most current model families are released with one.

Table 11: Paraphrases from the base model vs. the instruction-tuned sibling, Qwen2.5-7B.

	TriviaQA		NQ-Open		HotpotQA	
Method	EM	C-EM	EM	C-EM	EM	C-EM
Base model	38.36	49.81	19.20	26.68	17.85	21.17
CRAFT, base-model paraphrases	52.08	59.78	16.21	27.04	17.85	21.49
CRAFT, instruct paraphrases (default)	**58.18**	**60.70**	**24.32**	**27.20**	**21.44**	**22.65**

B.6 Does CRAFT Separate Near Neighbours?

Section 1 notes two routes to collapse. Questions can be close because their meanings are easily confused, or because they share sentence structure or a question template. In the second case, two questions can be highly similar without being confusable in any semantic sense. We therefore ran two checks, both with final-layer similarity on the TriviaQA validation set.

Nearest neighbours. For each question, we fix the 5% of questions most similar to it under the base model, and measure the mean similarity to this same set after CRAFT. It falls from 0.8134 to 0.5605, and it falls for 100% of questions.

Confusable pairs. We selected 100 pairs that are lexically near-identical (content-word Jaccard ≥ 0.6) but have disjoint gold answers, such as *"On a standard dartboard, which number lies between 4 and 6?"* and *"On a standard dartboard, what number lies opposite 18?"*. Their mean similarity falls from 0.8891 to 0.7616.

The effect is partly global. For the middle and the least similar 5% of each question's neighbours, similarity falls by 0.328 and 0.217, comparable to the 0.253 drop for the closest 5%. Together with §B.4 and §B.3, this suggests that CRAFT works through local paraphrase alignment together with global spreading, rather than by targeted repulsion of confusable neighbours.

B.7 Crowdedness Across Model Scale

The main text uses crowdedness percentiles, which are normalized within each model. Table 12 instead applies the same absolute threshold τ to the final-layer representations of every Qwen2.5 model and reports the median crowdedness. Crowdedness falls sharply from 1.5B to 7B, so in this range larger models do separate questions better. But the trend is not monotone: 32B is more crowded than 7B at every threshold. Crowdedness therefore does not depend on parameter count alone, and we do not yet know what else governs it. The negative association between crowdedness and accuracy still holds at 32B (Fig. 13).

Table 12: Median crowdedness $N_\tau(q)$ on the TriviaQA validation set at fixed thresholds τ, final-layer similarity.

τ	Qwen2.5-1.5B	Qwen2.5-3B	Qwen2.5-7B	Qwen2.5-32B
0.80	8157	7461	512	2665
0.85	4891	3138	33	604
0.90	750	169	1	13
0.95	3	1	0	0

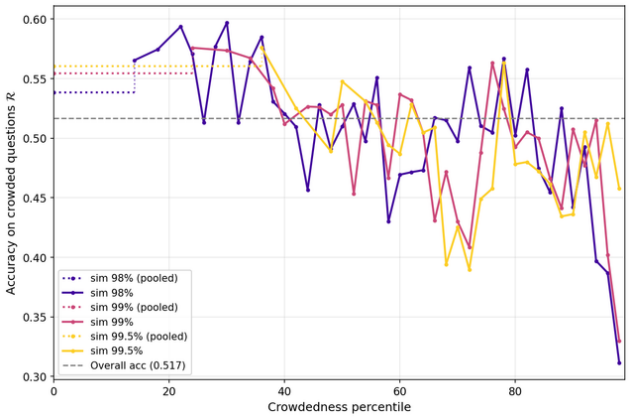


Figure 13: Accuracy on the crowded set $D_\tau(\pi)$ against crowdedness percentile π for base Qwen2.5-32B on TriviaQA, as in Fig. 1. Each curve uses a different similarity threshold τ (98th, 99th, and 99.5th percentile of final-layer similarities). Dotted segments on the left pool questions with too few neighbours to form a resolved bin, and the dashed line is overall accuracy (51.7%). As at smaller sizes, accuracy falls as crowdedness rises, from about 55–60% for the least crowded questions to about 31–46% for the most crowded, and stays mostly below overall accuracy from about the 40th percentile on.

B.8 Is the Gain Explained by Shorter Answers?

CRAFT makes answers much shorter (§A.1). We ran three controls to test whether this alone explains its gains. All use Qwen2.5-7B.

Accuracy at matched answer length. If the gains came only from shortening, questions whose base answers are already short should not improve. We bin questions by the length of the base model's answer (Table 13). 70% of questions have base answers of 1–2 words, and CRAFT still raises their EM by 12 points.

Table 13: TriviaQA accuracy binned by the length of the base model's answer, Qwen2.5-7B.

		Avg. length		EM		C-EM	
Base answer length	n	Base	CRAFT	Base	CRAFT	Base	CRAFT
1–2 words	6,976	1.41	1.56	49.50	61.91	51.20	63.43
3–5 words	1,024	3.43	3.02	34.96	51.95	41.50	53.91
6–15 words	1,058	10.45	2.22	1.32	50.28	50.38	55.95
16–46 words	903	22.89	2.91	0.00	46.40	49.50	53.49

Choosing the shortest of several samples. We sampled 10 answers per question from the base model and kept the shortest. This gives shorter answers than CRAFT, but lowers accuracy (Table 14, top).

Asking for a short answer. We prompted the base model with `Answer the following question in a short way: {question}\n:`. This hurts EM badly. It raises TriviaQA C-EM, but stays well below CRAFT (Table 14, bottom).

Table 14: Two ways of shortening the base model's answers, Qwen2.5-7B. *Top:* TriviaQA with average answer length (words). *Bottom:* all benchmarks.

Method	TriviaQA EM	TriviaQA C-EM	Avg. length
Base, greedy	38.36	49.81	4.53
Base, shortest of 10 samples	35.54	40.14	1.78
CRAFT	**58.18**	**60.70**	1.90

	TriviaQA		NQ-Open		HotpotQA	
Method	EM	C-EM	EM	C-EM	EM	C-EM
Base, short-answer prompt	18.65	53.82	13.05	25.65	8.70	17.85
CRAFT	**58.18**	**60.70**	**24.32**	**27.20**	**21.44**	**22.65**

B.9 Transfer and General Capabilities

We evaluated the Qwen2.5-7B CRAFT model, trained only on TriviaQA, without further tuning (Table 15). On WebQuestions (Berant et al., 2013), EM rises by 11.9 points, extending the transfer seen on NQ-Open and HotpotQA. TruthfulQA (Lin et al., 2022) changes very little; we read this as a check that factuality is not harmed, rather than as a broad gain. WikiText-103 (Merity et al., 2016) perplexity and MMLU (Hendrycks et al., 2021) accuracy are essentially unchanged, so CRAFT does not cause broad degradation of language modelling or knowledge-and-reasoning ability. These are safeguards, and do not show that every downstream capability is preserved. Since CRAFT is applied to base models, we do not evaluate instruction following.

Table 15: Transfer and general-capability checks, Qwen2.5-7B.

Evaluation	Base	CRAFT	Change
WebQuestions EM	17.08	28.94	$+11.86$
WebQuestions C-EM	34.50	35.93	$+1.43$
TruthfulQA MC1	25.95	26.56	$+0.61$
TruthfulQA MC2	40.83	41.20	$+0.37$
MMLU accuracy	74.33	74.21	-0.12
WikiText-103 perplexity (↓)	6.159	6.156	-0.003

B.10 Another Model Family: SmolLM3

To check that CRAFT is not specific to Qwen2.5 and Llama-2, we applied it to SmolLM3-3B, the released post-trained model rather than a base model, using final-layer similarity as for Qwen2.5. Instead of paraphrases from a SmolLM3 sibling, we reuse the paraphrases prepared for Qwen2.5-3B. Table 16 shows gains on every benchmark, largest in EM, so CRAFT also works with paraphrases from a different model family. Since this model is already post-trained, the result also suggests that CRAFT is not limited to base models.

	TriviaQA		NQ-Open		HotpotQA	
Method	EM	C-EM	EM	C-EM	EM	C-EM
SmolLM3-3B	33.06	46.62	4.46	23.41	8.14	16.76
SmolLM3-3B + CRAFT	**45.61**	**50.91**	**7.34**	**23.36**	**12.26**	**16.84**

Table 16: CRAFT on SmolLM3-3B, trained with paraphrases from the Qwen2.5-3B pipeline. Like Llama-2-7B, EM is computed on the first sentence/line, since SmolLM3-3B continues generating after its answer.

B.11 Statistical Resolution

A two-proportion chi-square test at the 5% level resolves accuracy differences of about 0.8, 1.4, and 1.0 points on TriviaQA, NQ-Open, and HotpotQA respectively. Most of CRAFT's gains over the base model and the strongest baselines in Table 1 exceed these margins. For the accuracy-vs-crowdedness curves, drops of 5 points are significant, and we observe drops of 20 points or more for very crowded questions.

B.12 Near-Duplicate Negatives

The TriviaQA training split contains repeated and near-repeated questions, and negatives exclude only the anchor itself. Telling a genuine paraphrase from a confusable but distinct question is hard, and any automatic rule will make errors in both directions. We can still bound the exposure. Over the 100k training pool as used in training (no de-duplication), the probability that a uniformly sampled negative is a non-identical near-duplicate of its anchor, at content-word Jaccard ≥ 0.8, is 7.6×10^{-7}. With four negatives per anchor, this gives about 0.3 expected such pairs among all $400{,}000$ negatives sampled during training. This estimate is conservative: many pairs it counts are not paraphrases at all. For example, *"Which element has the symbol Ag?"* and *"Fm is the symbol for which element?"* score above 0.8 but have different answers, and such confusable neighbours are exactly what CRAFT should separate.

C Implementation Details

C.1 Implementation Details of DoLa

We use the Hugging Face Transformers implementation of DoLa (Chuang et al., 2024). DoLa (High) contrasts the final layer against candidate layers from the upper half of the network, while DoLa (Low) uses candidate layers from the lower half. We used the repetition penalty as 1.2.

C.2 Implementation Details of ITI

We follow the original ITI (Li et al., 2024) implementation using TruthfulQA , adapting the dataset allocation to our setting since TruthfulQA (Lin et al., 2022) is not used as an evaluation benchmark to maximize the training dataset. Our procedure consists of two phases. In the first phase, we use the 20% of the dataset to for hyperparameter selection instead of the 10% used in the paper then use the remaining dataset to train the probes. No 2-fold procedure was used as we do not use TruthfulQA in our evaluations.

C.3 Implementation Details of Self-Eval-P(True)

We use the fully unsupervised pipeline of (Zhang et al., 2024) (Self-Eval-P(True)) with the same 100k TriviaQA training questions used for CRAFT. Since our training data consists of short-form atomic questions, we follow the paper's short-form generation task setup and skip the atomic part extraction step that relies on a larger model. For each TriviaQA question, we sample 20 candidate answers and compute confidence scores using the same prompting and probability scoring method as the paper, i.e., the Self-Eval-P(True) prompt that queries the model to assess the truthfulness of each candidate answer as $p(\text{True} \mid q, a)$, using the same model that is subsequently fine-tuned. Preferred and dispreferred pairs are then constructed from the scored candidates, and pairs with a preference score difference below 0.3 are discarded to avoid near-identical pairs, resulting in approximately 400k training pairs for Qwen2.5-7B model. DPO fine-tuning is then run for 3 epochs with a learning rate of 5×10^{-7}, $\beta = 0.1$, and an effective batch size of 16, which takes approximately 20 hours on 3 NVIDIA RTX 6000 Ada Generation GPUs for the Qwen2.5-7B model.

C.4 Implementation Details of FactTune-MC

Since our training questions are atomic by construction, the larger model atomic claim extraction step is skipped, consistent with our Self-Eval-P(True) setup. For each TriviaQA training question, we sample 20 candidate answers using the model that we are fine tuning and construct preferred and dispreferred pairs based on answer frequency, where more frequent answers are treated as more confident. Duplicate pairs are removed, and the top 5 pairs per question are retained based on the frequency gap between preferred and dispreferred answers, ensuring the training signal consists of pairs with the most clearly separable confidence scores. This results in approximately 320k training pairs for Qwen2.5-7B model. DPO fine-tuning is then run for 3 epochs with a learning rate of 5×10^{-7}, $\beta = 0.1$, and a batch size of 16, which takes approximately 18 hours on 3 NVIDIA RTX 6000 Ada Generation GPUs for the Qwen2.5-7B model.

C.5 Implementation Details of CRAFT

As described in the main text, the implementation of CRAFT consists of two stages:

1. Paraphrase and negative question generation
2. Fine-tuning with the objective defined in Equation 5

This section describes each stage in detail.

Stage 1: Paraphrase and Negative Question Generation. For the paraphrase generation step, we use the instruction-tuned version of each model that we subsequently fine-tune with CRAFT, namely Qwen2.5-1.5B-Instruct, Qwen2.5-3B-Instruct, Qwen2.5-7B-Instruct, and Llama-2-7B-Chat and prompt each model to generate paraphrases of the 100k TriviaQA training questions used for

CRAFT. For SmolLM3-3B, we reuse the Qwen2.5-3B paraphrases, so they come from a different model family than the model being fine-tuned (§B.10). We use three distinct prompt templates, each designed to encourage a different axis of lexical and structural variation:

System: `Rewrite the trivia question below. Output only the rewritten question. No explanation.`

Template 1 — Structural variation:
```
Change the grammatical structure of this question so it asks for the same answer:
{question}
```

Template 2 — Lexical variation:
```
Rewrite this question using different words. Same answer required:
{question}
```

Template 3 — Cross-lingual variation:
```
Translate this question to French. Keep proper nouns in English:
{question}
```

For each template, we sample $N = 5$ responses from the model with temperature $\tau = 0.8$ yielding a total of $3 \times 5 = 15$ paraphrase candidates per question.

Paraphrase Filtering via Self-Judging. Given the potentially variable quality of model-generated paraphrases, we filter candidates using the base model (the model that is subsequently fine-tuned) as a judge. Specifically, we prompt the base model with a binary verification query asking whether candidate question B seeks the *exact same* factual answer as the original question A:

System: `You are a data quality evaluator. Answer ONLY with Yes or No.`

User:
```
Does Question B ask for the EXACT SAME factual answer as Question A, even if worded differently?

Examples:
A: Who directed Inception?  B: Who was Inception's director?  → Yes
A: What is the capital of France?  B: What is the capital of Spain?  → No
A: Where in England was Judi Dench born?  B: Where in Wales was Judi Dench born?  → No

A: {original question}
B: {candidate paraphrase}
Answer:
```

We need to emphasize that the ground truth answers for the training questions are not used at any stage of this process. We measure the model's confidence in a positive judgment by extracting the log-probabilities of the first generated token. Concretely, for each candidate pair (A, B), we compute a confidence score:

$$s(A, B) = \frac{p_{\texttt{Yes}}}{p_{\texttt{Yes}} + p_{\texttt{No}} + \varepsilon}, \tag{8}$$

where $p_{\texttt{Yes}}$ and $p_{\texttt{No}}$ are obtained by summing the exponentiated log-probabilities over all surface-form variants of affirmative and negative tokens respectively (e.g., `"Yes"`, `" Yes"`, `"yes"`, and their negative counterparts). A candidate paraphrase is retained only if $s(A, B) \geq 0.80$.

Prior to judge scoring, we apply a set of lightweight heuristic pre-filters to discard degenerate outputs. A candidate is rejected if any of the following conditions hold:

- It contains fewer than 4 words.
- It contains non-Latin Unicode characters.
- It begins with a filler phrase (e.g., *"Sure"*, *"Of course"*, *"Here is"*).
- It does not take the syntactic form of a question (i.e., it neither contains a question mark nor begins with an interrogative word such as *who*, *what*, *where*, *when*, *which*, *how*, or *why*).
- Its word-overlap ratio with the original question exceeds 0.95, indicating a near-duplicate.

After filtering, we retain up to $k = 4$ paraphrases per question, ranked by their confidence score $s(A, B)$.

Negative Sampling. Rather than generating hard negatives with the model, we sample them directly from the TriviaQA training pool. For each anchor question, we randomly draw $N_{\text{neg}} = 4$ questions from the full set of 100k training questions, excluding the anchor itself. Because these questions are drawn from a large and topically diverse trivia corpus, they are very likely unrelated to the anchor (see §B.12), providing a reliable source of negatives for the contrastive objective in Equation 5.

Stage 2: Fine Tuning with CRAFT. Following the data generation and filtering process, we fine-tune the base versions of the models (not the Instruct ones) using the objective defined in Equation 5. The hidden states of the anchor questions, paraphrases, and negatives are extracted from the last token position of the following prompt:

```
Question: {question}
Answer:
```

We again emphasize that the ground-truth answers are not used at any point during this stage.