Fortifying Hallucination Detection to Out-of-Domain Data

Anonymous ACL submission

Abstract

Hallucinations remain one of the major barriers to the reliable deployment of Large Language Models (LLMs). Recent works have explored hallucination detection using both supervised classification based approaches and unsupervised metric based approaches, with the latter remaining popular since they do not require labeled data. In this work, we study the performance and generalizability of these approaches through a comprehensive empirical analysis, and introduce a simple approach that improves the generalizability of supervised hallucination detectors by training on a curated, multi-domain data mix, which can complement the subsequent addition of task-specific data. In our experiments on hallucination detection on 697K QA samples from 11 open source QA datasets, we show that incorporating this general training allows supervised methods to surpass unsupervised metric based methods by an average of +7.25% on out of domain data, without addition of any task-specific data. We also analyze scaling behaviors and estimate how much task-specific data is required to achieve reliable performance, finding that models augmented with general data require up to 40.3% less task-specific data to achieve close to optimal performance. Together, our findings highlight a simple path toward fortifying supervised hallucination detectors against domain shift.

1 Introduction

Large language models' (LLMs) tendency to generate factually incorrect or misleading outputs, often termed *hallucinations*, poses a critical barrier to their adoption in high-stakes or out-of-distribution settings (Kim et al., 2025; Dahl et al., 2024). Detecting hallucinations at test time is challenging (Sahoo et al., 2024). One naive approach is to fact-check outputs against an external reference or database (Chern et al., 2023; Min et al., 2023). However, this requires costly retrieval and fails when no ground-truth reference exists, calling for other methods to detect hallucinations.

One line of research has leveraged *uncertainty estimates* as potential signals of model reliability, and thus, for hallucination detection (Farquhar et al., 2024). *Unsupervised* methods analyze output probabilities, response consistency, or cluster semantic entropy (Abdaljalil et al., 2025; Farquhar et al., 2024; Nikitin et al., 2024; Venhuizen et al., 2019), or use linear probing (Kossen et al., 2024), while *supervised* methods train classifiers on model states to detect hallucinations (Liu et al., 2024) (Figure 1). While supervised methods outperform unsupervised metrics in-domain (Liu et al., 2024), they lose robustness out-of-domain, often underperforming unsupervised methods when faced with distribution shift. We empirically validate this in §4.1 by benchmarking supervised against unsupervised approaches across multiple data domains and model families, showing that while supervised methods excel in-domain, their performance degrades sharply under domain shift, often underperforming unsupervised methods.

Motivated by this, we introduce an approach to bridge this performance-generalization gap by training a supervised model on a large and heterogeneous dataset to yield more robust hallucination detection, adapting domain generalization and pretraining works that show how diverse training data can improve out-of-domain robustness in other tasks (e.g., NLI, NER and sentiment analysis; Hosseini et al., 2024; Stacey et al., 2025; Yu et al., 2022) to uncertainty-based hallucination detection. In addition, we study scaling law type behaviors, investigating how much domain-specific data is needed to obtain a classifier with acceptable performance. We conduct our experiments using different dataset splitting strategies, iteratively testing combinations of train-test splits across a comprehensive dataset of 697K QA samples from 12 open source QA datasets, which we further categorize

into broad domains. Our evaluation spans 10 different LLMs of varying sizes to ensure robustness across model families.

Our main contributions are:

- We present the first comprehensive benchmarking of supervised hallucination detection methods against state-of-the-art unsupervised metric-based approaches, across a wide range of domains and language models, finding that unsupervised methods outperform supervised ones out-of-domain but not in-domain.
- We introduce a novel data-driven strategy for improving out-of-domain robustness of supervised hallucination detectors by scaling up heterogeneous training data, demonstrating its effectiveness at narrowing the performance-generalization gap.
- We provide the first systematic analysis of the relationship between training data volume and classifier performance, offering practical guidelines on data requirements for effective uncertainty-based hallucination detection.

2 Background & Related Works

Background: Hallucination Detection via Uncertainty Quantification Broadly, there have been two styles of approaches for uncertainty quantification. On one hand, *unsupervised* methods generally seek to estimate uncertainty via analyzing output probabilities, consistency and similarity across multi-generations (Abdaljalil et al., 2025; Farquhar et al., 2024; Nikitin et al., 2024) and via linear probing (Kossen et al., 2024). For example, semantic entropy (Venhuizen et al., 2019) clusters multiple generated responses to estimate semantic entropy, where prompts leading to diverse sets of clusters implies a higher level of uncertainty. On the other hand, work has explored *supervised* classifiers with internal language model states as features trained to estimate uncertainty (Liu et al., 2024). Figure 1 demonstrates this approach in the white-boxing setting, where internal states of the LLM are used to predict labeled hallucinations. This approach also applies to black-box and gray-box settings. For example, in the gray-box setting, we can restrict ourselves to the use aggregated statistics based on the output logits of the generating LLM. In addition, when combined with a supervised model that produces probability estimates, these probabilities can be interpreted as uncertainty estimates.

Benchmarking UQ and Hallucination Detection Previous efforts to benchmark uncertainty quantification have focused on examining performance vs. efficiency (in terms of number of generations needed) tradeoffs of UQ methods requiring multiple generations (Xiong et al., 2024; Valentin et al., 2024). Ye et al. (2024) utilizes UQ metrics and various datasets to benchmark the certainty level of different LLMs and calibratedness of their responses. Xiong et al. (2023); Tian et al. (2023) provide extensive benchmarks across different datasets and models but focus mainly on verbalized uncertainty, and do not benchmark on in-domain vs out-domain. Lastly Liu et al. (2024) benchmarks supervised uncertainty quantification methods across methods and studies the extent of domain transfer across datasets, but does not focus on when such domain transfer occurs.

Out-Of-Domain Robustness Approaches to improving out-of-domain robustness in machine learning include training and optimization-based techniques (Wang et al., 2022; Yang et al., 2021) as well as data-centric approaches. In computer vision, increasing diversity of training data and data augmentation have been shown to reduce overfitting and improve generalization (Zhou et al., 2020; Rahman et al., 2019). In NLP, synthetic data have been explored as a means to enhance domain generalization in natural language inference (Hosseini et al., 2024). Building on these insights, we are the first to study whether large and heterogeneous pretraining can play a similar role for hallucination detection under domain shift.

3 Approach & Experimental Setup

Our goal is to develop a generalizable method for hallucination detection in the question-answering (QA) setting. We adopt a supervised classification approach, showing that it outperforms alternative metric based methods, but that its performance degrades out of domain. To address this challenge, we explore training on a mix of in-domain and heterogeneous data. This section introduces the task setup and supervised detection approach as well as datasets, models and experimental setup.

Hallucination in QA setting While ultimately our goal is to measure hallucinations in any LLM output, as a starting point, we focus mainly on the QA setting, where a LLM generates a textual response to an input question. This response is

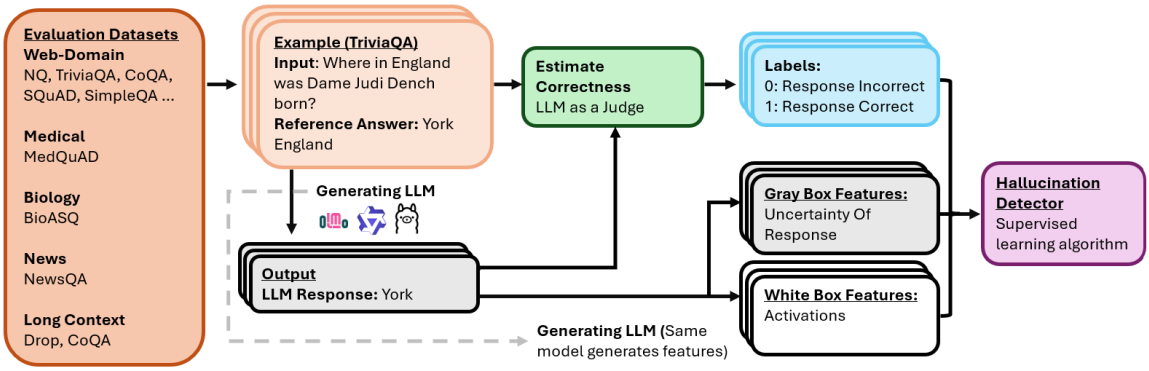


Figure 1: Overview of supervised learning based hallucination detection pipeline. We use examples from QA style benchmarks to generate positive and negative examples of when LLM an hallucinates. Along with generated features based on the prompt and LLM response, a downstream supervised learning algorithm is applied to train a classifier that detects these hallucinations based on the input features. We have two types of data, a general training mix and a task specific set, both of which we use to train the model, the task specific set confers in-domain specificity while the general training mix works to prevent overfitting, promoting robustness under domain shift.

scored against a reference response, and incorrect responses are treated as hallucinations. The task is then to detect such hallucinations given the input question and the corresponding LLM response. We assume a white-box setting with full access to model internals (e.g., activations, output logits).

We view performance in short-form QA as foundational for hallucination detection in longer-form generative tasks such as abstractive summarization. This is evidenced by many long-form methods that first extract atomic facts as a pre-processing step (Thirukovalluru et al., 2024; Min et al., 2023; Kadavath et al., 2022), making fact-level hallucination detection essential. The QA setting is also advantageous as it is relatively straightforward to measure correctness, whereas long-form outputs require fact-extraction and verification across many sentences, a capability that is still an active area of research (Chen et al., 2025; Liu et al., 2025; Wei et al., 2024).

Our Approach Our approach focuses on training a multi-domain supervised hallucination detector using white-box and grey-box LLM features (Figure 1). The pipeline has three components: (1) generating candidate answers from the *generating LLM* on QA datasets (2) constructing feature representations from their internal activations and (3) training a classifier to distinguish between hallucinated and correct responses. Formally, each sample consists of features $(X_{\text{prompt}}, X_{\text{response}})$ paired with a binary label $y \in \{0, 1\}$, where $y = 1$ denotes a hallucinated (incorrect) response and $y = 0$ a correct one.

For features, we follow prior work (Liu et al., 2024; Azaria and Mitchell, 2023) and extract activations from the middle and final layers of the generating LLM, using only activations corresponding to the last token of both the prompt and generated response. We then apply dimensionality reduction using truncated SVD, necessary because the activations are high dimensional (3584-4096) and training data is limited for some of our splits. Our experiments focus on how data can affect performance, thus for consistency, we apply SVD across all models even when dataset size exceeds the feature dimension.

We evaluate classifiers of the form

$$f(X_{\text{prompt}}, X_{\text{response}}) = y_{\text{score}}, \quad y_{\text{score}} \in [0, 1], \tag{1}$$

where y_{score} denotes the predicted score of hallucinations, with high scores indicating a higher likelihood of hallucination. Final predictions are obtained by thresholding y_{score}, although we also evaluate quality of the score itself as described in §3.2. In our experiments, we use a Random Forest classifier, following prior work in Liu et al. (2024). Random Forests are widely regarded as a strong and robust baseline across many different problem settings (Wainer, 2016), offering good performance with minimal tuning (Probst et al., 2019). We also experimented with other classifiers, namely XGBoost (Chen and Guestrin, 2016) and penalized linear regression and found Random Forest to be very competitive with these. Additionally we keep hyperparameter constant throughout all experiments (Appendix E). While optimal hyperparameter vary

across dataset splits, we adopt a single setting to avoid exhaustive tuning, further our focus is on in-domain vs out-domain performance which we find to be largely insensitive to hyperparmeter choice.

3.1 Datasets, Models and Methods

Evaluation Datasets We collect data from 12 different QA style benchmarks across different domains and input context length. These datasets along with their domains are listed in Table 1. Several of the datasets come with a train and a test set. For all our tested models, we found that the performance on both these sets tended to be similar, and thus we use both. Combined this gives us a dataset of 697K examples.

Table 1: Benchmarking Datasets used in our study.

Domain	Benchmarks	Size
Encyclopedic	TriviaQA, NQ, HotpotQA, SQuAD	363.8k
KB QA	SimpleQA, WebQuestions	10.1k
Medical/Bio	bioASQ, MedQuAD	21.1k
Conversational	CoQA	116.6k
Reasoning	DROP	83.6k
News	NewsQA	78.3k
Various	OpenLLM	23.7k

LLMs Considered Our evaluation spans three model families: Llama-3.1 (Dubey et al., 2024), OLMo2 (OLMo et al., 2024) and Qwen2.5 (Team, 2024). Additionally we consider various post training schemes, Tulu 3 (Lambert et al., 2024) and SimPO-based (Meng et al., 2024) for Llama-3.1, and the post training process in (Mu et al., 2025) for Qwen-2.5. We evaluate models across different training stages, namely SFT and SFT+DPO. For computational purposes, we primarily use the 7-8B variants of these models except for OLMo2 where we considered both 7B and 32B models.

Baseline Unsupervised Methods We consider a mix of multi generation and single generation methods. These are Semantic Entropy (Farquhar et al., 2024; Venhuizen et al., 2019), Sindex (Abdaljalil et al., 2025), GNLL (Aichberger et al., 2024) and PTrue (Kadavath et al., 2022). We elaborate more on these methods in Appendix A.

3.2 Evaluation Metrics

We evaluate performance using two metrics. For threshold-agnostic evaluation, we report Area Under Receiver Operating Curve (AUROC), which measures how well a score differentiates between positive and negative examples across all decision thresholds. AUROC has been used extensively in previous work (Liu et al., 2024; Aichberger et al., 2024; Farquhar et al., 2024). For threshold-aware evaluation, we use F1-score, optimizing thresholds on validation data to maximize this metric. Additional details on these and alternative metrics are provided in Appendix B.

3.3 LLM As a Judge

We use an LLM as a judge to determine correctness, as prior work has shown that alternative metric-based approaches (e.g. ROUGE, BLUE, BERTScore) can yield inaccurate labels and substantially alter results (Santilli et al., 2024; Ielanskyi et al., 2025; Janiak et al., 2025). We adopt `Qwen3-14B` (Yang et al., 2025) as our primary judge model, finding that it showed strong agreement with internal human annotation as well as with more expensive models such as such as `gpt-4o` and `gpt-4o-mini`, while being significantly more cost efficient (Appendix C).

4 Results

Towards building a generalizable hallucination detection system we first quantify the degree of supervised performance loss out-of-domain (§4.1). Then, we demonstrate the effectiveness of our generalized heterogeneous dataset to reduce domain gap (§4.2). Finally, we investigate scaling law type behaviors, namely how performance scales with number of data samples (§4.3). Additionally, Appendix H provides a theoretical analysis of GE data's contribution to out-of-domain robustness.

4.1 How do supervised methods perform against unsupervised methods in both the in-domain and out-of-domain setting?

We compare supervised and unsupervised methods in both the in-domain and out-of-domain settings across the 11 of the 12 benchmark datasets, excluding the openLLM dataset due to its highly heterogeneous content. For each source dataset, performance is evaluated on held-out source data (in-domain) and all other datasets (out-of-domain), repeating this for all source-target pairs and generating LLMs. Figure 2 reports these results macro-averaged across datasets. In addition, we report performance gaps and associated error bars that account for dataset-specific variations in Appendix F.

As expected, supervised methods consistently outperform unsupervised ones in-domain across

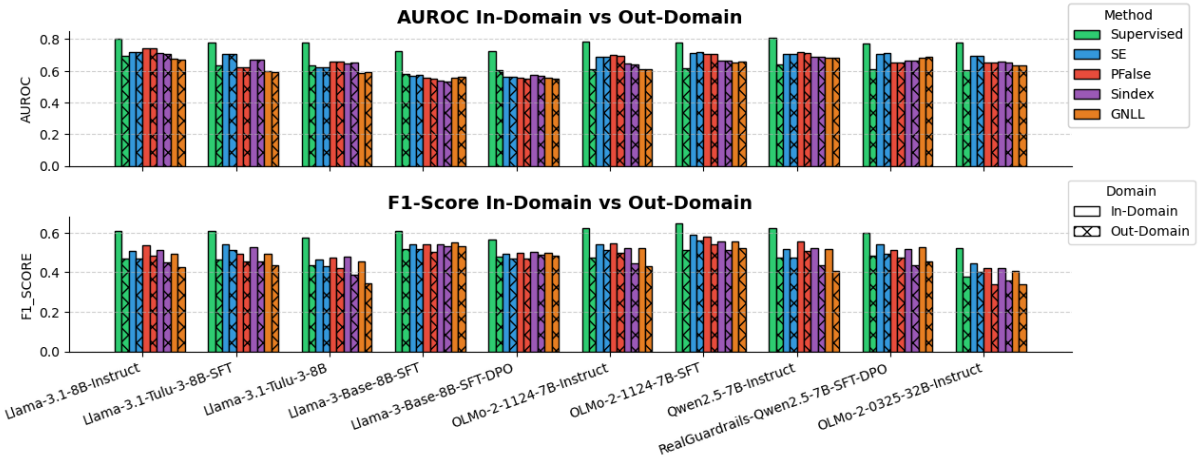


Figure 2: Aggregated In-domains and Out-of-domain evaluations for various models macro-averaged across 11 different datasets. Solid bars represent in-domain performance and cross-hatched represents out-domain performance. Top Row: AUROC, Bottom Row: F1-Score

all benchmarks (Figure 2), with this advantage being consistent across model types and evaluation metrics. **Out-of-domain, however, supervised models experience substantial performance drops**, often losing their in-domain advantage and performing on par with or even below unsupervised metrics, highlighting their sensitivity to distribution shift. Among unsupervised methods, the multi-generation based metrics, SE, Sindex and PFalse generally outperform GNLL, which uses a single generation. Performance varies across generating LLMs, with SE often the top performer, consistent with prior work (Farquhar et al., 2024) though other studies report different rankings (Abdaljalil et al., 2025), likely due to differences in generating LLM or evaluation setup.

4.2 Can training on a large general data-mix mitigate out of domain performance degradation?

Next, we explore our primary question of how dataset diversity affects generalization. We consider a dataset level leave-one-out split, where one dataset is held out as the target, termed the **Task-Specific (TS)** dataset. The remaining datasets are combined into a larger diverse **General (GE)** dataset for training. This setup allows us to study whether increased training diversity can mitigate the performance drop observed when moving from in-domain to out-of-domain evaluation.

Leave-one-out experiments. We first evaluate our approach using a dataset level leave-one-out data split, for this we iteratively select one dataset to leave out as the target dataset. The remaining datasets then function as the general set. The target (TS) dataset is split into a train, test and validation dataset, this allows us to assess the hallucination detection capabilities under two settings, 1) where the classifier is trained solely on the GE set 2) where the classifier is trained on both the GE set and some data from the target domain / TS set.

Figure 3, displays results aggregated over the heldout-target datasets for the different LLMs. We highlight three key observations:

First, **supervised GE models generally outperform unsupervised methods**. Comparing GE-only to unsupervised baselines, Figure 3 shows that GE-only generally matches or exceeds the best unsupervised methods across models. This advantage is especially pronounced in AUROC, where GE-only achieves higher average performance than all unsupervised approaches. However, the pattern is less consistent for F1-score: for `OLMo2-Instruct`, `Qwen2.5-Instruct`, and `Llama3.1-Tulu-SFT`, GE-only underperforms. A closer analysis for `OLMo2-Instruct` (Figure 4) reveals that this dip stems from poor performance on longer-context datasets DROP and CoQA. Since AUROC remains high, this suggests some sensitivity to thresholding on the general set. Similar patterns are observed for the other models.

Second, we examine the value of target-specific data. Across all held-out targets and generating LLMs, classifiers trained exclusively on general (GE) data consistently under-perform compared to those trained with task-specific (TS) data, underscoring the important of target-specific supervi-

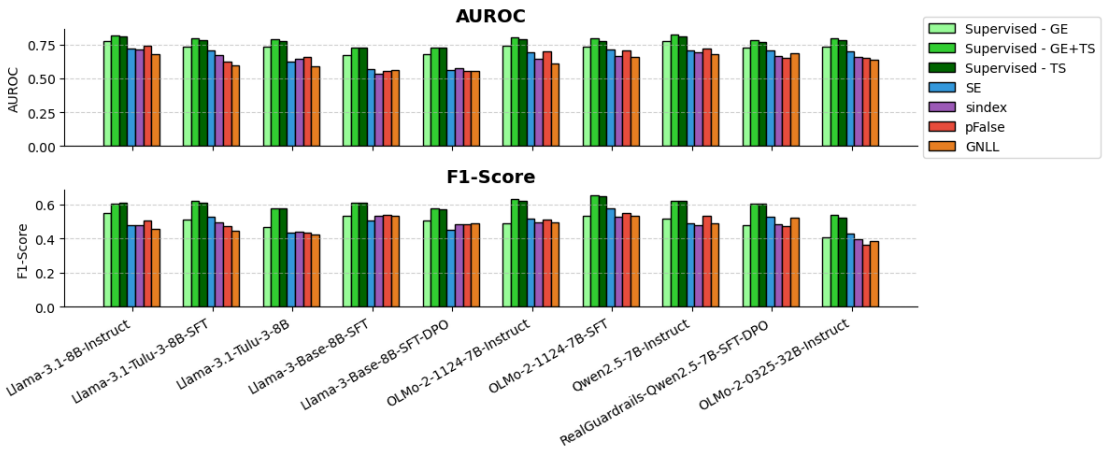


Figure 3: AUROC and F1-Score of different methods aggregated across heldout-benchmarks. Groups represent the detection performance for different generating models. Bars represent the method used with green bars showing different variants of supervised methods (differing by training data used) and remaining bars representing unsupervised methods. You can shorten model names to add space

sion. This result is unsurprising given our earlier findings on the effectiveness of supervised methods in-domain. Interestingly, the gap between GE-only and TS-only models persists even for target datasets with strong similarities to others in the GE set, these would include target datasets such as Natural Questions. We have observed in the single source experiments, that in some cases training on another source can outperform a training on the target source. Thus the consistent overall gap suggests a degree of negative transfer when relying on heterogeneous GE data. Nevertheless, when comparing expected performance of single source on out-of-domain, GE-only still seems to provides a net benefit.

Finally, GE+TS offers limited benefits over TS when target data is abundant. After averaging across heldout target datasets, we find that GE+TS slightly outperforms TS-only by only a small margin. We hypothesize that target dataset size explains this. In particular the target datasets tested are generally quite large (on the order of tens of thousands of examples). In such cases, classifiers may already be saturated by task-specific data, limiting additional performance gains when adding the general examples. We explore these scaling effects more systematically in later sections.

Broad domain shifts While dataset-wise leave-one-out splits are a standard way to assess out of domain generalization, many of the evaluation datasets in our suite share substantial domain overlap with one another, for example NQ, TriviaQA both draw heavily from Wikipedia based sources, and thus we expect that domain transfer from them will be strong. To better disentangle this effect we curate GE sets based on 'broad' domains that are more dissimilar to one another, which we list in Table 1. These results are shown in Figure 5 which plot results of GE and GE+TS under different choices of the GE set. We only evaluate for two evaluation datasets, bioASQ and MedQuAD which have the most dissimilar domain in our set of evaluation benchmarks.

Across these settings we observe the same trend that using a GE set can result in a classifier with better performance than one that uses unsupervised metric based methods, this results stays consistent over several choices of GE sets and across models. We find that with the largest "Encyclopedic-Wiki" domain included, we did not see much variations in performance when including other domains.

Overall, these results show that the addition of a large heterogeneous general training mix provides a strong foundation for supervised methods, surpassing unsupervised approaches on AUROC. Although this alone is general training alone is insufficient to match the performance achieved with target-specific data, we find complementary effects when they are used together.

4.3 Scaling laws: how much in domain data is needed to adapt classifier?

Motivated by the promising performance of classifiers trained using GE + TS data, we investigate how much in-domain data labeled for hallucinations is needed to train a classifier with adequate

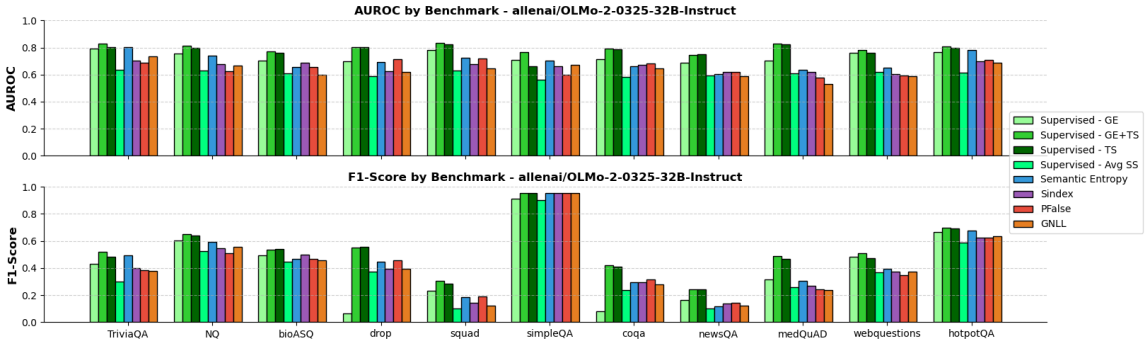


Figure 4: AUROC of supervised and unsupervised hallucination detector on *OLMo2-32B-Instruct*. Each group represents the heldout target dataset. Bars represent the method used with green bars showing different variants of supervised methods (differing by training data used) and remaining bars representing unsupervised methods.

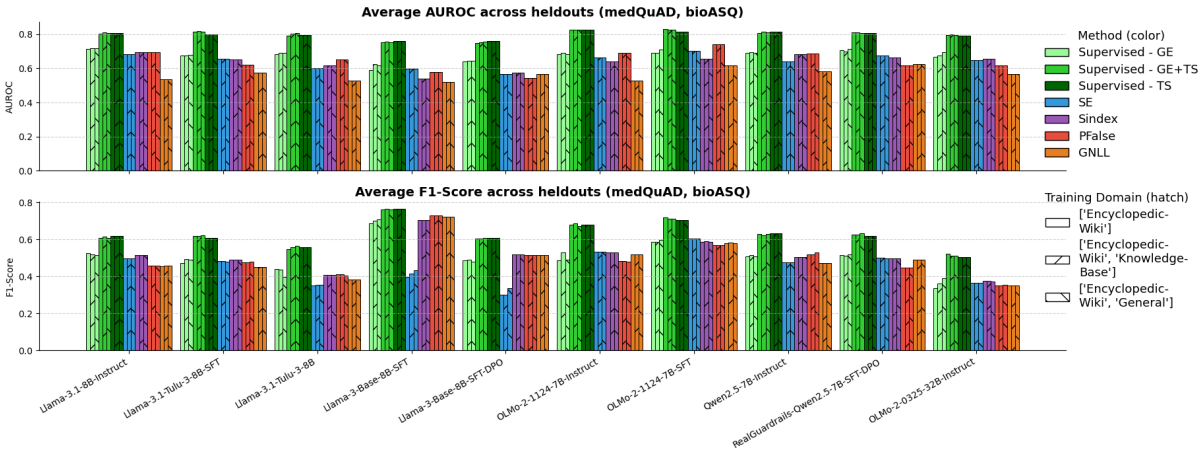


Figure 5: AUROC and F1-scores for aggregated over two target datasets - MedQuAD and bioASQ under different choices of general data domain. Bars represent the method used with green bars showing different variants of supervised methods (differing by training data used) and remaining bars representing unsupervised methods. Hatching patterns on the bar denominate the choice of general training domain when applicable.

performance, and whether using a large general dataset can improve data efficiency. Obtaining labeled hallucination data can be challenging, making it desirable to supplement limited in-domain labeled data with a larger general training mix. To examine these effects, we plot learning curves (Perlich, 2011) under both the TS and GE+TS setting. These curves plot the test-set performance as we vary the amount of task specific training data, highlighting how quickly a classifier can learn to accurately detect hallucinations. Here, we consider only BioASQ and MedQuAD (as done earlier), which are the most unlike the other datasets. Figure 6 plots an example of these learning curves on `Llama 3.1 - Instruct`. **Generally, GE+TS dominates TS only in the low data regime of < 1000 examples, with TS-only catching up soon after**. This trend stays consistent across the 10 LLMs tested, indicating that general-domain data is most beneficial when task-specific supervision is limited. The learning curves further show that GE+TS consistently matches or outperforms TS only when given the same number of task-specific sample, and we therefore recommend GE+TS as a robust default training setup.

To gain further actionable insights, we seek estimates of two quantities from these learning curves:

1. **Crossover Point**: At how many training examples does the TS-setting outperform the GE-only setting.

2. **Saturation Point**: At what sample count do we achieve 95% of detection performance in either the GE+TS or TS setting.

The crossover point between TS-only and GE-only marks the estimated amount of data where the use of target specific data outperforms the use of the general set. This highlights a trade-off between cost

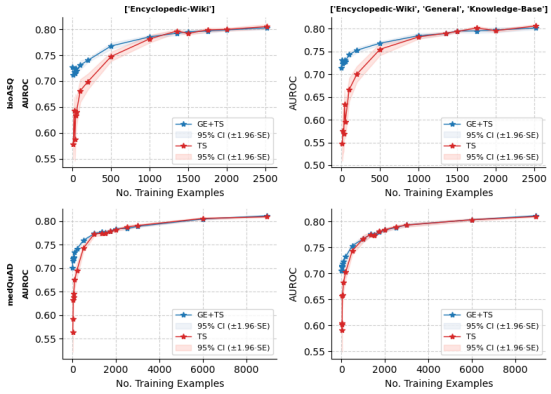


Figure 6: Learning Curves for `Llama-3.1-Instruct` under TS only and GE+TS for different amounts of target specific data. Top Row: BioASQ as target domains. Bottom Row: MedQuAD as target domains. Lines represent average AUROC value across 10 random seeds and shaded area represents 1.96 * SE.

of annotation and model performance. A crossover point at a low number of samples suggests that supervision from the target domain is highly valuable, whereas one at a high number of samples suggests that the general set itself might be the most practical choice unless a large and labeled target/task-specific dataset is available. The saturation points give some guidance on how many labeled samples are needed in order to maximize the performance of a hallucination detection model.

Table 2: Performance on held-out datasets for cross-over experiments. Top: mean, Bottom: (std dev).

Data	N	Cross-over GE/TS	Sat. Pt. GE+TS	Sat. Pt. TS
BioASQ	2520	372.9 (128.6)	485.1 (201.9)	820.7 (263.0)
MedQuAD	8350	348.4 (149.6)	1168.0 (306.0)	1408.2 (712.4)

Table 2 summarizes estimates for the both the crossover and saturation point. Across both heldout datasets, the expected cross-over between GE and TS occurs at roughly 310-350. Saturation point is reached relatively early for BioASQ, at 465 samples for GE+TS, and later for MedQuAD, at 1100 for GE+TS. We believe this is largely a function of the total number of samples we had for the experiment, in the case of medQuAD, the higher number of theoretical samples we had pushed the estimated maximal performance level higher, resulting in later saturation point. Consistently across both datasets, saturation point for GE+TS is much lower than that for TS only, 40.3% lower for bioASQ and 17.0% for MedQuAD, showing that use of the GE set improves data efficacy. These results guide practitioners on whether additional labeling is worthwhile, suggesting diminishing returns from excessive task-specific data when using GE+TS. For example, in the case of MedQuAD the expected saturation point of GE+TS at 1168 indicates that if one were to label an additional 7182 data points (8350-1168), they would only expect about a 5% increase in performance on AUROC.

4.4 Other key results

We further explore other aspects of our method in the appendix. Appendix I studies scaling the size and composition of the GE set. Performance peaks around $N = 50000$ examples, and domain proximity to the target dataset is beneficial when TS data is unavailable. Appendix J applies our methods to other hallucination detectors HaMI (Niu et al., 2025) and TSV (Park et al., 2025), showing that we can make similar gains in robustness.

5 Conclusion

In this paper we introduce a data-driven approach to make supervised hallucination detectors robust under domain shift. We first showed that supervised hallucination detection methods significantly outperforms unsupervised approaches in the in-domain setting, but that this advantage disappears in the out-domain setting, where unsupervised metric based approaches are comparatively more robust. We showed that the use of general, heterogeneous data that need not be in the same domain as the target domain can provide a useful foundation for training supervised classifiers, with such classifiers generally surpassing unsupervised models even when data in the target domain is unavailable. Moreover our scaling experiments show that incorporating such general data improves data efficiency when combined with target specific data, as classifiers require fewer target-specific samples to achieve the same performance. These findings demonstrate that supervised UQ-based hallucination detection methods remain a valuable tool. Practitioners can apply classifiers trained on large general datasets and expect performance that exceeds unsupervised approaches, further, when available, incorporating target-specific data to these classifiers further improves performance, consistently outperforming unsupervised methods.

Limitations

This work considers only English datasets and meaning that detected hallucinations are only in English. Thus the findings may not generalize to other languages. Our experiments primarily focus on smaller scale models (Up to 8B parameters), with limited analysis of a single 32B model (OLMo2). We do not evaluate frontier scale models (eg 70B parameters and above). In addition, our evaluation is restricted to short form question answer with open ended responses, and thus the results may not extend to longer form generation settings.

References

Samir Abdaljalil, Hasan Kurban, Parichit Sharma, Erchin Serpedin, and Rachad Atat. 2025. Sindex: Semantic inconsistency index for hallucination detection in llms. *arXiv preprint arXiv:2503.05980*.

Lukas Aichberger, Kajetan Schweighofer, and Sepp Hochreiter. 2024. Rethinking uncertainty estimation in natural language generation. *arXiv preprint arXiv:2412.15176*.

Amos Azaria and Tom Mitchell. 2023. The internal state of an llm knows when it's lying. *arXiv preprint arXiv:2304.13734*.

Shai Ben-David, John Blitzer, Koby Crammer, Alex Kulesza, Fernando Pereira, and Jennifer Wortman Vaughan. 2010. A theory of learning from different domains. *Machine learning*, 79(1):151–175.

Leo Breiman. 2001. Random forests. *Machine learning*, 45(1):5–32.

Mingda Chen, Yang Li, Xilun Chen, Adina Williams, Gargi Ghosh, and Scott Yih. 2025. Factory: A challenging human-verified prompt set for long-form factuality. *arXiv preprint arXiv:2508.00109*.

Tianqi Chen and Carlos Guestrin. 2016. Xgboost: A scalable tree boosting system. In *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, pages 785–794.

I Chern, Steffi Chern, Shiqi Chen, Weizhe Yuan, Kehua Feng, Chunting Zhou, Junxian He, Graham Neubig, Pengfei Liu, and 1 others. 2023. Factool: Factuality detection in generative ai–a tool augmented framework for multi-task and multi-domain scenarios. *arXiv preprint arXiv:2307.13528*.

Matthew Dahl, Varun Magesh, Mirac Suzgun, and Daniel E Ho. 2024. Large legal fictions: Profiling legal hallucinations in large language models. *Journal of Legal Analysis*, 16(1):64–93.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, and 1 others. 2024. The llama 3 herd of models. *arXiv e-prints*, pages arXiv–2407.

Sebastian Farquhar, Jannik Kossen, Lorenz Kuhn, and Yarin Gal. 2024. Detecting hallucinations in large language models using semantic entropy. *Nature*, 630(8017):625–630.

Mohammad Javad Hosseini, Andrey Petrov, Alex Fabrikant, and Annie Louis. 2024. A synthetic data approach for domain generalization of nli models. *arXiv preprint arXiv:2402.12368*.

Mykyta Ielanskyi, Kajetan Schweighofer, Lukas Aichberger, and Sepp Hochreiter. 2025. Addressing pitfalls in the evaluation of uncertainty estimation methods for natural language generation. In *ICLR Workshop: Quantify Uncertainty and Hallucination in Foundation Models: The Next Frontier in Reliable AI*.

Denis Janiak, Jakub Binkowski, Albert Sawczyn, Bogdan Gabrys, Ravid Schwartz-Ziv, and Tomasz Kajdanowicz. 2025. The illusion of progress: Re-evaluating hallucination detection in llms. *arXiv preprint arXiv:2508.08285*.

Saurav Kadavath, Tom Conerly, Amanda Askell, Tom Henighan, Dawn Drain, Ethan Perez, Nicholas Schiefer, Zac Hatfield-Dodds, Nova DasSarma, Eli Tran-Johnson, and 1 others. 2022. Language models (mostly) know what they know. *arXiv preprint arXiv:2207.05221*.

Yubin Kim, Hyewon Jeong, Shan Chen, Shuyue Stella Li, Mingyu Lu, Kumail Alhamoud, Jimin Mun, Cristina Grau, Minseok Jung, Rodrigo Gameiro, and 1 others. 2025. Medical hallucinations in foundation models and their impact on healthcare. *arXiv preprint arXiv:2503.05777*.

Jannik Kossen, Jiatong Han, Muhammed Razzak, Lisa Schut, Shreshth Malik, and Yarin Gal. 2024. Semantic entropy probes: Robust and cheap hallucination detection in llms. *arXiv preprint arXiv:2406.15927*.

Nathan Lambert, Jacob Morrison, Valentina Pyatkin, Shengyi Huang, Hamish Ivison, Faeze Brahman, Lester James V Miranda, Alisa Liu, Nouha Dziri, Shane Lyu, and 1 others. 2024. Tulu 3: Pushing frontiers in open language model post-training. *arXiv preprint arXiv:2411.15124*.

Linyu Liu, Yu Pan, Xiaocheng Li, and Guanting Chen. 2024. Uncertainty estimation and quantification for llms: A simple supervised approach. *arXiv preprint arXiv:2404.15993*.

Xin Liu, Lechen Zhang, Sheza Munir, Yiyang Gu, and Lu Wang. 2025. Verifact: Enhancing long-form factuality evaluation with refined fact extraction and reference facts. *arXiv preprint arXiv:2505.09701*.

Yishay Mansour, Mehryar Mohri, and Afshin Rostamizadeh. 2009. Domain adaptation: Learning bounds and algorithms. *arXiv preprint arXiv:0902.3430*.

Yu Meng, Mengzhou Xia, and Danqi Chen. 2024. Simpo: Simple preference optimization with a reference-free reward. *Advances in Neural Information Processing Systems*, 37:124198–124235.

Sewon Min, Kalpesh Krishna, Xinxi Lyu, Mike Lewis, Wen-tau Yih, Pang Wei Koh, Mohit Iyyer, Luke Zettlemoyer, and Hannaneh Hajishirzi. 2023. Factscore: Fine-grained atomic evaluation of factual precision in long form text generation. *arXiv preprint arXiv:2305.14251*.

Norman Mu, Jonathan Lu, Michael Lavery, and David Wagner. 2025. A closer look at system prompt robustness. *arXiv preprint arXiv:2502.12197*.

Alexander Nikitin, Jannik Kossen, Yarin Gal, and Pekka Marttinen. 2024. Kernel language entropy: Fine-grained uncertainty quantification for llms from semantic similarities. *Advances in Neural Information Processing Systems*, 37:8901–8929.

Mengjia Niu, Hamed Haddadi, and Guansong Pang. 2025. Robust hallucination detection in llms via adaptive token selection. *arXiv preprint arXiv:2504.07863*.

Team OLMo, Pete Walsh, Luca Soldaini, Dirk Groeneveld, Kyle Lo, Shane Arora, Akshita Bhagia, Yuling Gu, Shengyi Huang, Matt Jordan, and 1 others. 2024. 2 olmo 2 furious. *arXiv preprint arXiv:2501.00656*.

Seongheon Park, Xuefeng Du, Min-Hsuan Yeh, Haobo Wang, and Yixuan Li. 2025. Steer llm latents for hallucination detection. *arXiv preprint arXiv:2503.01917*.

Claudia Perlich. 2011. Learning curves in machine learning. In *Encyclopedia of machine learning*, pages 577–580. Springer.

P Probst, MN Wright, and AL Boulesteix. 2019. Hyperparameters and tuning strategies for random forest, wires data mining and knowledge discovery, 9, e1301.

Mohammad Mahfujur Rahman, Clinton Fookes, Mahsa Baktashmotlagh, and Sridha Sridharan. 2019. Multi-component image translation for deep domain generalization. In *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 579–588. IEEE.

Pranab Sahoo, Prabhash Meharia, Akash Ghosh, Sriparna Saha, Vinija Jain, and Aman Chadha. 2024. A comprehensive survey of hallucination in large language, image, video and audio foundation models. *arXiv preprint arXiv:2405.09589*.

Andrea Santilli, Miao Xiong, Michael Kirchhof, Pau Rodriguez, Federico Danieli, Xavier Suau, Luca Zappella, Sinead Williamson, and Adam Golinski. 2024. On a spurious interaction between uncertainty scores and answer evaluation metrics in generative qa tasks. In *Neurips Safe Generative AI Workshop 2024*.

Joe Stacey, Lisa Alazraki, Aran Ubhi, Beyza Ermis, Aaron Mueller, and Marek Rei. 2025. How to improve the robustness of closed-source models on nli. *arXiv preprint arXiv:2505.20209*.

Qwen Team. 2024. Qwen2 technical report. *arXiv preprint arXiv:2407.10671*.

Raghuveer Thirukovalluru, Yukun Huang, and Bhuwan Dhingra. 2024. Atomic self-consistency for better long form generations. *arXiv preprint arXiv:2405.13131*.

Katherine Tian, Eric Mitchell, Allan Zhou, Archit Sharma, Rafael Rafailov, Huaxiu Yao, Chelsea Finn, and Christopher D Manning. 2023. Just ask for calibration: Strategies for eliciting calibrated confidence scores from language models fine-tuned with human feedback. *arXiv preprint arXiv:2305.14975*.

Simon Valentin, Jinmiao Fu, Gianluca Detommaso, Shaoyuan Xu, Giovanni Zappella, and Bryan Wang. 2024. Cost-effective hallucination detection for llms.(2024). *URL https://arxiv. org/abs/2407.21424*.

Noortje J Venhuizen, Matthew W Crocker, and Harm Brouwer. 2019. Semantic entropy in language comprehension. *Entropy*, 21(12):1159.

Jacques Wainer. 2016. Comparison of 14 different families of classification algorithms on 115 binary datasets. *arXiv preprint arXiv:1606.00930*.

Jindong Wang, Cuiling Lan, Chang Liu, Yidong Ouyang, Tao Qin, Wang Lu, Yiqiang Chen, Wenjun Zeng, and Philip S Yu. 2022. Generalizing to unseen domains: A survey on domain generalization. *IEEE transactions on knowledge and data engineering*, 35(8):8052–8072.

Jerry Wei, Chengrun Yang, Xinying Song, Yifeng Lu, Nathan Hu, Jie Huang, Dustin Tran, Daiyi Peng, Ruibo Liu, Da Huang, and 1 others. 2024. Long-form factuality in large language models. *Advances in Neural Information Processing Systems*, 37:80756–80827.

Miao Xiong, Zhiyuan Hu, Xinyang Lu, Yifei Li, Jie Fu, Junxian He, and Bryan Hooi. 2023. Can llms express their uncertainty? an empirical evaluation of confidence elicitation in llms. *arXiv preprint arXiv:2306.13063*.

Miao Xiong, Andrea Santilli, Michael Kirchhof, Adam Golinski, and Sinead Williamson. 2024. Efficient and effective uncertainty quantification for llms. In *Neurips Safe Generative AI Workshop 2024*.

An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, and 1 others. 2025. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*.

Shuai Yang, Kui Yu, Fuyuan Cao, Lin Liu, Hao Wang, and Jiuyong Li. 2021. Learning causal representations for robust domain adaptation. *IEEE Transactions on Knowledge and Data Engineering*, 35(3):2750–2764.

Fanghua Ye, Mingming Yang, Jianhui Pang, Longyue Wang, Derek Wong, Emine Yilmaz, Shuming Shi, and Zhaopeng Tu. 2024. Benchmarking llms via uncertainty quantification. *Advances in Neural Information Processing Systems*, 37:15356–15385.

Yu Yu, Shahram Khadivi, and Jia Xu. 2022. Can data diversity enhance learning generalization? In *Proceedings of the 29th international conference on computational linguistics*, pages 4933–4945.

Kaiyang Zhou, Yongxin Yang, Timothy Hospedales, and Tao Xiang. 2020. Learning to generate novel domains for domain generalization. In *European conference on computer vision*, pages 561–578. Springer.

A Unsupervised Methods

Semantic Entropy Semantic entropy estimates the uncertainty of a language model by measuring the diversity of meanings across multiple generations. For a single prompt p we generate S different responses $c_1, ..., c_S$. Semantic entropy clusters these responses into semantic equivalent clusters through the use of a Natural language inference model that checks bidirectional entailment.

Let $\mathcal{C} = \{C_1, \ldots, C_K\}$ denote the resulting set of semantic clusters. The probability mass of a cluster can be obtained by summing the probabilities of the generated samples assigned to that cluster:

$$P(C_k \mid p) = \sum_{c_i \in C_k} P(c_i \mid p). \quad (2)$$

Semantic entropy computed as the entropy over the distribution of semantic clusters:

$$\mathrm{SE}(p) = -\sum_{k=1}^{K} P(C_k \mid p) \log P(C_k \mid p). \quad (3)$$

Higher SE implies uncertainty and a higher likelihood of the response being hallucinated.

Sindex Sindex similar to semantic entropy, also measures semantic consistency across multiple generations. However, instead of using entailment from a NLI model to form clusters, Sindex uses sentence embeddings combined with a hierarchical clustering algorithm based on cosine similarity.

Let $\mathcal{C} = \{C_1, \ldots, C_K\}$ denote the resulting clusters and let p_i be the proportion of generations assigned to cluster C_i. Sindex computes an adjusted entropy:

$$\mathrm{SINDEx}(p) = -\sum_{i=1}^{K} p'_i \log p'_i, \quad (4)$$

where the adjusted cluster weight incorporates intra-cluster semantic coherence:

$$p'_i = p_i \cdot s_i, \quad (5)$$

and s_i is the average pairwise cosine similarity between embeddings of responses in cluster C_i.

PTrue PTrue is an LLM as a judge approach that queries the generating LLM on whether a given statement is True or False, this is done by appending a question to the statement and measuring the generated token probability of the *true* token. Following (Kadavath et al., 2022) we use a variation of PTrue where we pass in multiple candidate generations in the prompt as well as the main response (for which we score correctness). Additionally, for hallucination detection we actually need the probability inverse (1-pTrue) score, which we call pFalse.

GNLL GNLL is a likelihood-based score designed to estimate aleatoric uncertainty. (Aichberger et al., 2024) show that under a 0-1 loss, the negative log-likelihood of the most likely output sequence is a good estimate of the aleatoric uncertainty. While this quantity is hard to identify due to computational intractability of the LLM generating space we can approximate this using either beam search decoding or greedy decoding. For this work we use the NLL of the greedy decoded sequence.

B Evaluation Metrics

Threshold-Agnostic Metrics Prior works on UQ and hallucination detection primarily evaluate performance using Area Under Receiver Operating Curve (AUROC) ((Liu et al., 2024; Aichberger et al., 2024; Farquhar et al., 2024)). AUROC measures how well a score differentiates positive and negative examples. In our casem the score is either a UQ metric or the probability score generated by a supervised hallucination detector. As a threshold-agnostic metric, AUROC evaluates performance

across all decision thresholds. Other options in this category are Area Under Precision Recall Curve (AUPRC) and Area Under Accuracy-Reject Curve (AUARC), but we primarily report performance using AUROC to keep consistent with previous work.

Threshold-Aware Metrics Many unsupervised metric-based methods output unbounded scores, this is the case for 3 out of 4 methods, SE, Sindex and GNLL. For them to be used practically we have to set thresholds for determining what are hallucinated responses. With these thresholds set, we can evaluate standard classification metrics such as accuracy, precision and recall. For this work, in order to optimize the threshold we use maximize F1-score on a separate validation set. To ensure fairness in comparison, we optimize thresholds for unsupervised methods on the training set, while for supervised detectors we rely on a separate validation subset.

C LLM as a Judge

Here we give more details about the LLM-as-a-Judge Procedure. Broadly, we first evaluated GPT-4 as a judge model and found that its assessments were well aligned with human annotations, though on a relatively small sample. Given the cost of using GPT-4 at scale, we sought an open-source alternative with similar reliability. Among several candidates, we identified `Qwen3-14B` (Yang et al., 2025), which exhibited the highest inter-annotator agreement with GPT-4. We therefore adopt `Qwen3-14B` as our primary judge model in this work (see Appendix C.2 for details.)

C.1 Labeling

To generate labels for hallucination detection, we use a 3 class system:

- **0 - Non-Hallucinations** For generated responses deemed correct by the LLM judge
- **1 - Hallucinations** For generated responses deemed in-correct by the LLM judge
- **2 - Non Responses** For generated responses deemed as non-response by the LLM judge

We construct a small human-labeled set of 25 examples across datasets and model family, including brief annotations of why they are labeled the way they are. These examples are used in a few-shot prompt (25-shot) to guide the LLM judge in labeling the full set of responses, this prompt is given in Figure 7.

Table 3: Inter-annotator agreement between LLM-as-a-judge models and human raters.

Model	Cohen's Kappa	Accuracy vs Human
GPT-4o	0.84	92%
Qwen3-14B	0.85	93%

C.2 Open-Source Models

Our initial efforts used mainly `gpt-4o` as a judge, but we found this prohibitively expensive for the number of experiments we wanted to do, thus we sought an open-source alternative that would perform as well. We mainly tested models from the Qwen3 family, named the 14B transformer and the 30B MoE model. We found Cohen's Kappa to be higher for the 14B transformer as shown in Table 3. We also annotate by hand a sample of a 100 model completions and compare that with our models, which is shown in Table 3, for this sample GPT-4o and Qwen3-14B perform nearly identical. Further both have high accuracy and inter annotator agreement, further validating our choice to use Qwen3-14B.

Lastly, as a final check, we label an additional N=315 data points and compare Qwen3-14B to human accuracy. For this we sample across 3 LLMs, and focus one 5 task, with extra samples given to bioASQ and MedQuAD, which are more specialised domains. The results are shown in Table 5 and Table 6. We found that agreement remains high across benchmark datasets, including more domain specific ones such as BioASQ (90.0%) and MedQuAD (88.9%), suggesting no strong evidence of systematic bias on more specialized QA tasks. We also observe consistent performance across model families (91.4–93.3% accuracy), indicating that the judge is not disproportionately biased towards a particular model's outputs.

D LLMs Tested

E Supervised Training Details

Here we furnish addition details on the training procedure and model use.

LLM-as-a-Judge Prompt

You are an expert evaluator for question-answering systems. You will be assessing the quality of answers to a given question.
Task: Determine if the candidate answer contains the correct factual information to answer the question.
Rules: - Respond with `0` if the candidate answer is equivalent in meaning to any reference answer (synonyms/context OK), or if it contains the specific correct answer. - Respond with `1` if the candidate contains wrong facts, repeats the question, provides no answer, has the wrong entity, or differs from all reference answers. - Respond with `2` if the model does not give an answer, asks a clarifying question, or refuses to answer. - Accept additional context around correct answers. - Accept geographic or temporal equivalents. - Reject if the core answer is missing or incorrect.
Instructions: Reference answers may be in a numbered list. Score `0` if the candidate matches any reference answer.
Evaluation Examples:

- Q: What movie starred Tom Cruise? Ref: Top Gun Candidate: "Top Gun starring Tom Cruise" → 0
- Q: What movie starred Tom Cruise? Ref: Top Gun Candidate: "Brad Pitt was in Top Gun" → 1
- Q: What's the name of Mob's brother's Spanish VA in Mob Psycho 100? Ref: Javier Olguín Candidate: "I do not have information on the Spanish voice actor..." → 2
- Q: What is the major difficulty in carrying out the plan? Ref: Improving the relationship between Taiwan and the mainland Candidate: "I'd be happy to help you identify potential difficulties..." → 2

Figure 7: LLM-as-a-Judge prompt used for labeling responses from different datasets. Only 4 few-shot examples are shown here due to space constraints, but for actual applications we use a 25-shot example

Table 4: Inter-annotator agreement between LLM-as-a-judge models and GPT-4o.

Model	Cohen's Kappa vs GPT-4o	Accuracy vs GPT-4o
GPT-4o	1.00	100%
Qwen3-14B	0.756	85.4%
GPT-4o-Mini	0.808	89.4%
Qwen3-A3B30B	0.659	82.2%

Table 5: Agreement between Qwen3-14B judgments and human labels across benchmarks.

Benchmark	N	Accuracy vs Human Labels (Qwen3-14B)
BioASQ	90	90.0%
MedQuAD	90	88.9%
DROP	45	91.1%
SimpleQA	45	100%
SQuAD	45	97.7%

E.1 Dataset

Table 8 gives additional information on dataset sizes and the splits to create the train test and validation sets per evaluation benchmark.

E.2 Supervised Hallucination Classifier

The supervised learning model used as the classifier is a Random Forest (Breiman, 2001). We did not do extensive hyperparameter tuning per generating LLM and dataset due to compute constraints, instead opting for a setting of 100 trees and with remaining settings being the default in scikit-learn. We also included an additional dimensionality reduction step since for many of our experiments we have the case where the dimensionality of the features exceeded the number of training examples. This dimensionality reduction is carried out using Singular Value Decomposition (SVD), for which we use the implementation in scikit-learn. We set a fix dimensionality of 300 after SVD, which is

Table 6: Agreement between Qwen3-14B judgments and human labels across evaluated models.

Model	N	Accuracy vs Human Labels (Qwen3-14B)
Llama 3.1 8B Instruct	105	91.4%
Qwen 2.5 7B Instruct	105	93.3%
OLMo 2 32B Instruct	105	92.3%

then used in the Random Forest. Lastly as a preprocessing step before SVD we apply standard scaling to the raw features.

Features Following (Liu et al., 2024) we use activations from the middle and last layer of the model. For each of these layers we take the activation value corresponding to both the last token and the prompt, this creates a large feature space as we take a total of 4 activations per input, for example if the model's hidden size is 4096 then the size of the features corresponding to this activation is 16384 (4*4096). In addition we explored the use of several probability based features but find that they did not impact performance much and omitted them.

Additional Classifiers We additionally benchmark two additional classifier models, XGBoost and feed-forward neural networks and find comparable performance to Random Forests suggesting that the observed generalization trends are largely classifier agnostic.

F RQ1 Additional Figures

G RQ2 Additional Figures

H Domain Specificity of Various Classifiers

H.1 Domain Shift and Error Decomposition

Here we provide further analysis on the different behaviors of GE, TS and GE+TS based hallucination detectors. We first consider the hypothetical error decomposition under domain shift (Ben-David et al., 2010; Mansour et al., 2009). Let D_s and D_t denote the source and target distributions. Then where $\epsilon_t(h)$ denotes the target error of hypothesis $h \in \mathcal{H}$, we have:

$$\epsilon_t(h) \leq \epsilon_s(h) + \frac{1}{2} d_{\mathcal{H}\Delta\mathcal{H}}(D_s, D_t) + \lambda \quad (6)$$

The first term $\epsilon_s(h)$ denotes the error on the source domain. $\frac{1}{2} d_{\mathcal{H}\Delta\mathcal{H}}(D_s, D_t)$ is a divergence term that measures how different source and target distributions are in the feature space induced by h. The third term λ is a hypothesis mismatch term which captures aggregate performance of the best hypothesis $h^* \in \mathcal{H}$ in both source and target domain. Interpreting our three training regimes TS,GE and GE+TS within this framework we hypothesize that:

- **TS Trained Classifiers** minimize $\epsilon_s(h)$, but have high divergence $d_{\mathcal{H}\Delta\mathcal{H}}(D_s, D_t)$ due to the encoding or use of domain-specific features.
- **GE Trained Classifiers** trained on heterogeneous datasets in contrast, should create a feature representation that reduces $d_{\mathcal{H}\Delta\mathcal{H}}(D_s, D_t)$, although at the cost of increasing source specific error $\epsilon_s(h)$ on any specific dataset.
- **GE+TS Trained Classifiers** are likely to have a favorable balance of both terms.

We have seen that TS trained hallucination detectors generally have much higher performance than GE trained variants, validating that these models have a lower source specific error. To explain why TS trained classifiers tend to fail out of domain, we seek to validate whether they indeed rely more heavily on domain-specific features.

H.2 Domain Classification Probes and Experimental Setup

To obtain a proxy that is compatible with the discrete feature sets used in our random forest classifiers, we exploit the Gini importance based feature ranking produced during training to train domain classification probes. Our hypothesis is that classifiers trained on task-specific data will utilize features that better encode domain-specific artifacts, resulting in better performance on the domain identification task over GE and GE+TS feature sets.

For each training regime (TS, GE, GE+TS) we extract the top 10% of features in the hallucination detector model. We then train a downstream classifier whose goal is not to detect hallucinations, but rather to predict from which data set a sample came from. This is done by training a binary domain classification probe for every task-specific dataset D_{TS}, where we set the positive class as samples from D_{TS} and the negative class as samples from D_{GE}. To evaluate we look at the F1-score relative to the F1-score obtained by a domain classifier trained on all available features.

Table 7: Models evaluated in our study.

Hugging Face Model name	Model family	Size	Training stage
meta-llama/Llama-3.1-8B-Instruct	Llama-3.1	8B	Instruct
allenai/Llama-3.1-Tulu-3-8B-SFT	Llama-3.1	8B	SFT
allenai/Llama-3.1-Tulu-3-8B	Llama-3.1	8B	Instruct
princeton-nlp/Llama-3-Base-8B-SFT	Llama-3	8B	SFT
princeton-nlp/Llama-3-Base-8B-SFT-DPO	Llama-3	8B	SFT + DPO
allenai/OLMo-2-1124-7B-Instruct	OLMo-2	7B	Instruct
allenai/OLMo-2-1124-7B-SFT	OLMo-2	7B	SFT
Qwen/Qwen2.5-7B-Instruct	Qwen2.5	7B	Instruct
normster/RealGuardrails-Qwen2.5-7B-SFT-DPO	Qwen2.5	7B	SFT + DPO
allenai/OLMo-2-0325-32B-Instruct	OLMo-2	32B	Instruct

Table 8: Benchmarking Datasets used in our study.

Benchmark	Domain	Total Size	Train Size	Test Size
TriviaQA	Encyclopedic	76 523	53 566	22 957
NQ	Encyclopedic	91 535	64 074	27 461
bioASQ	Biology	4 719	3 303	1 416
CoQA	Conversational	116 630	81 641	34 989
DROP	Reasoning-heavy	83 625	58 538	25 086
HotpotQA	Encyclopedic	97 846	68 492	29 354
MedQuAD	Medical	16 407	11 485	4 922
NewsQA	News	78 372	54 860	23 512
SimpleQA	KB QA	4 326	3 028	1 298
SQuAD	Encyclopedic	98 061	68 643	29 418
WebQuestions	KB QA	5 810	4 067	1 743
OpenLLM	Various	23 839	16 687	7 152

H.3 Results

Figure 18 displays the relative F1-Score for 5 task-specific datasets bioASQ, MedQUAD, SimpleQA, CoQA and NewsQA, the GE set chosen is the 'encyclopedic-wiki' set which consists of datasets with encyclopedic like content (Table 1). The aggregated results show that **TS trained classifiers tend to prioritize features which are domain specific**, generally achieving higher relative F1-Score over GE and GE+TS selected feature sets. Figure 19 displays the same results but disaggregated to their individual LLMs. We see that the trend is consistent, holding for almost all tested datasets in 10 of the 12 models.

These results provide empirical evidence explaining the domain shift problem under TS regime, and why GE and GE+TS training helps mitigate this. TS-only models may have low bias on source domain, but experience large domain shifts. In contrast GE only models may have higher bias as they miss task-specific nuances experience a smaller domain shift effect. Lastly we have seen that GE+TS balances both the bias and domain shift term.

I Fine-Grained Analysis of General Set Construction

In this section we detail additional analysis investigating the role of GE data in training hallucination detectors. We focus on three questions

- How does domain proximity between training and evaluation data affect classifier generalization?
- Does proximity-based curation of GE sets impact downstream performance?
- How does GE set size and composition interact?

These experiments are conducted on two more specialised domains, bioASQ and medQuAD, which represent challenging out-of-distribution settings. We select three model families (`meta-llama-3.1-8b`, `qwen25-7b-instruct`, `olmo2-7b-instruct`) to ensure findings are not model-specific.

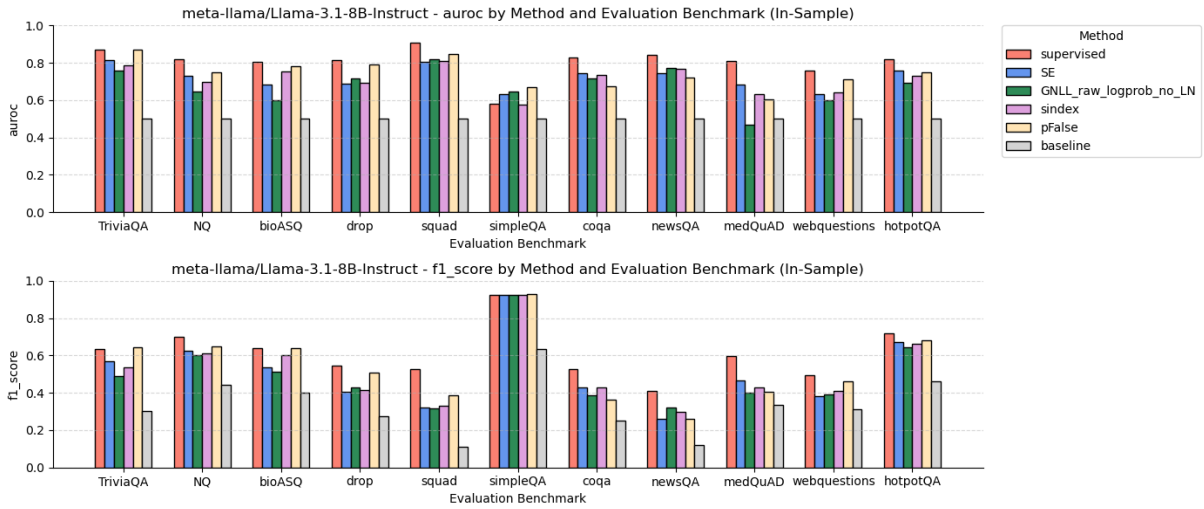


Figure 8: In-domains evaluations of "Llama-3.1-Instruct" across 11 different datasets. Supervised methods plotted along with 4 unsupervised methods and one random classifier baseline. Top Row: AUROC, Bototm Row: F1-Score

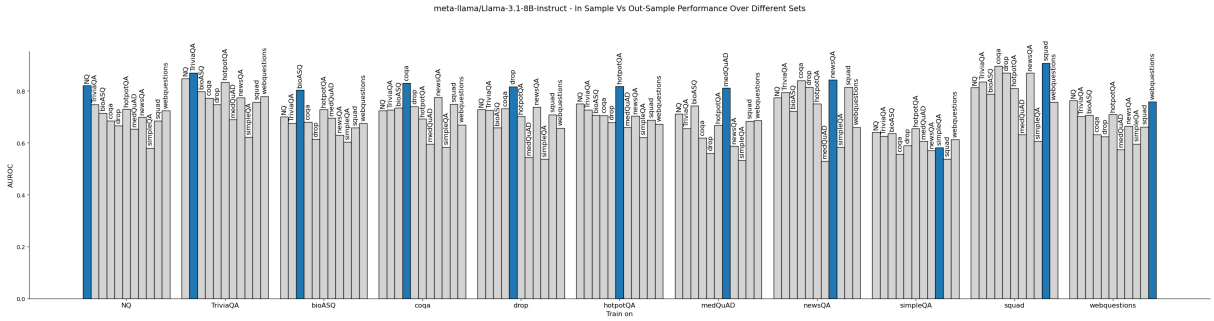


Figure 9: In-domains evaluations for various models macro-averaged across 11 different datasets. In order to benchmark supervised methods against unsupervised methods we pick the top performing unsupervised method for macro-averaging. Top Row: AUROC, Bottom Row: F1-Score

I.1 Domain Proximity as a Proxy for GE Set Quality

To understand what makes a GE set effective for OOD hallucination detection, we first examine the transferability of single-source trained classifiers across all dataset pairs, producing the AUROC heatmaps shown in Figure 12.

Observation 1: We observe heterogeneity in domain transferability. The heat maps reveal substantial heterogeneity in cross-domain transfer performance:

- In-domain (diagonal) entries consistently achieve the highest AUROC across all three model families.
- We also observe that off diagonal performance shows some correlation with semantic similarity between datasets; for example, bioASQ and medQuAD transfer well to each other but poorly to open domain datasets like simpleQA or drop.

Motivated by this, we study the use of single-source transfer AUROC as a proxy for domain similarity when constructing GE sets.

I.2 Proximity-Based GE Set Construction and its Effect on Performance

Using the computed single-source transfer AUROCs for ranking, we construct ordered GE sets as follows. For a given target evaluation set (eg, bioASQ or medQuAD), we rank all source datasets by their single-source transfer AUROC. We then slide a window of size 4 across this ranked list, thus each window position yields a GE set whose summed AUROC increases monotonically. Gives us GE sets that range from the least domain proximal, to the most domain proximal combination of single sources. In addition, we cap all GE sets at $N = 20000$ examples to remove confounding

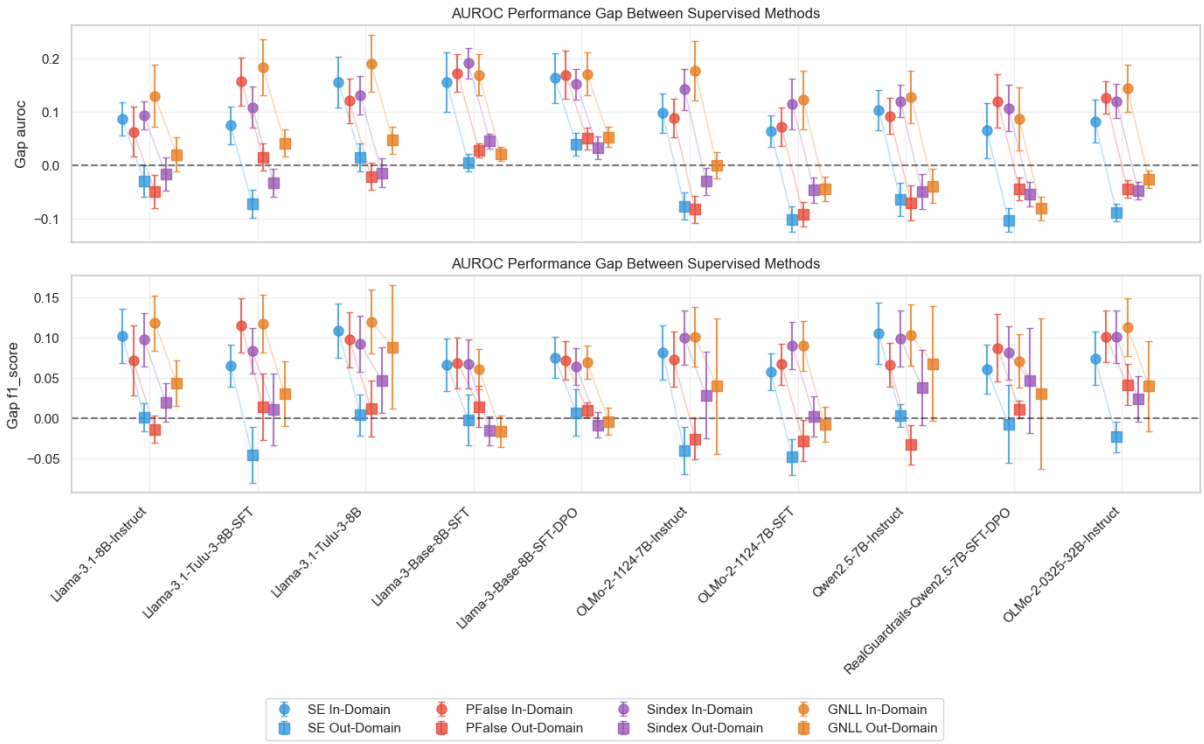


Figure 10: Difference plots comparing unsupervised methods to supervised methods. Heights of marker correspond to the performance difference between supervised methods and unsupervised methods. Top: AUROC, Bottom: F1-Score. Positive values indicate that supervised methods performance bettter than unsupervised method. Circles represent in-domain and squares represent out-of-domain. Error Bars correspond to 1.96 * Standard Error calculated over scores from aggregated source-target pairs.

effects of data volume. We then get results on the target dataset under both the GE and the GE+TS setting. Results are shown in Figure 13.

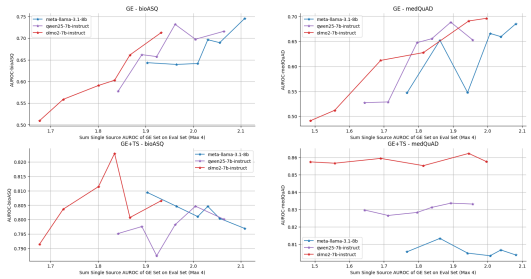


Figure 13: AUROC on bioASQ and medQuAD for Different GE sets with increasingly large summed AUROC. Plotted under GE (Top) and GE+TS (Bottom) settings across three model families. Higher x-axis values correspond to more domain similar GE sets.

Observation 2: GE set composition matters, but only without TS data.

- In the **GE-only** setting, downstream performance tracks the window ordering closely across all three model families. Higher-proximity GE sets consistently outperform lower-proximity ones on both bioASQ and medQuAD. This validates the use of single-source transfer AUROC as a signal to curate GE sets and suggests that GE sets which are most similar to the target domain tend to perform better.
- In the **GE+TS** setting, this relationship disappears. We see that lower-aggregated AUROC GE sets match higher ones once target-specific data is introduced, suggesting that fine-tuning compensates for mismatches in GE set composition.

These results indicate that domain proximity is critical as a design criterion when labeled target data is scarce or unavailable.

I.3 Effect of GE Set Size on Performance

We next study how GE set size interacts with classifier OOD performance by varying the number of training examples from 5K to 275K across 4 fixed domain splits. Results are shown in Figure 14.

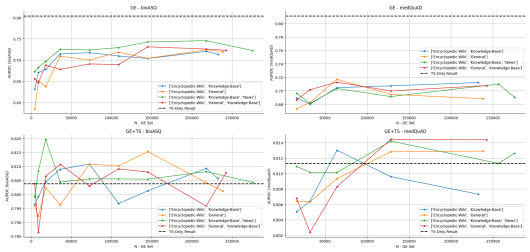


Figure 14: AUROC on bioASQ and medQuAD as a function of GE set size across four fixed domain splits, in both the GE-only (Top) and GE+TS (Bottom) settings. The dashed line denotes the TS-only baseline.

Observation 3: Performance gains from GE set size plateau quickly.

- In the **GE-only** setting, performance improves substantially in the 0-50K range across all domain-splits but plateaus quickly after, suggesting that the initial scale-up captures most of the learnable signal from general data.
- In the **GE+TS** setting, results are more mixed: adding 0-50K GE examples generally helps, but adding too many can somtimes cause negative transfer, likely due to the growing imbalance between general and target-specific signal.

These results suggest that practitioners can achieve near ceiling GE performance with only around $N = 50000$ examples.

I.4 Analysing Proximity Based Curation vs GE Set Size

Finally, we directly compare a curated Top 4 SS GE set (containing four most domain-proximal sources), a Bottom 4 SS GE set (four least proximal), and a larger un curated diverse baseline across varying dataset sizes. Results are shown in Figure 15.

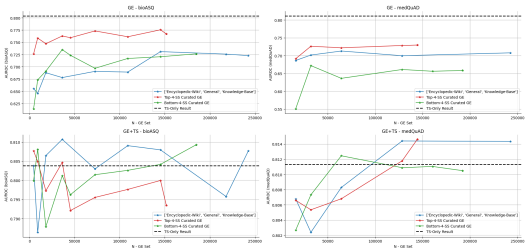


Figure 15: AUROC on bioASQ and medQuAD comparing Top-4-SS curated, Bottom-4-SS curated, and un-curated diverse GE sets across varying dataset sizes, under GE-only (Top) and GE+TS (Bottom). In the GE-only setting, Top-4-SS consistently outperforms the diverse baseline, while Bottom-4-SS consistent under performs. This advantage disappears in the GE+TS setting, where the diverse uncurated set performs on par or better. Dashed line denotes the TS-only baseline.

Observation 4: GE set curation matters more than data scaling. But only in the GE only setting.

- In the **GE-only** setting, the Top-4-SS curated set maintains a consistent performance advantage over the diverse uncurated baseline across all dataset sizes, demonstrating that proximity-based curation is beneficial even when the uncurated set has access to substantially more data. The Bottom-4-SS set performs persistently worse, indicating that poor curation carries a lasting penalty that additional data cannot overcome.
- In the **GE+TS** setting, this advantage disappears, with the diverse uncurated set performing on par or better than the Top-4-SS set. The disadvantage we saw in the Bottom-4-SS also diminishes.

J Comparing to Other Supervised Hallucination Detection Approaches

J.1 Alternative Hallucination Detection Approaches

In addition to our activation-based probe model, we evaluate two recent hallucination detection approaches: **Truthfulness Separator Vector (TSV)** (Park et al., 2025) and **HaMI** (Niu et al., 2025).

TSV learns a steering vector from labeled examples that captures a direction in activation space associated with correct or incorrect responses. At inference time, hallucination detection is performed by measuring the alignment between model activations and the learned signal direction.

Instead of classifying using activations from a fixed token (eg. the last token), HaMI accounts for activations across all generated tokens, automatically training the model to select informative tokens using a multiple instance learning based training procedure.

Our method differs from both approaches in its simplicity. To reiterate we extract the activation of the **final token of the prompt and generated response** and train a **Random Forest (RF) classifier** on this representation, similar to activation probing approaches such as (Azaria and Mitchell, 2023). Comparatively, TSV requires repeated forward passes through the language model during training, while HaMI requires extracting activations from all generated tokens.

J.2 Experimental Setup

Since both TSV and HaMI are substantially more computationally expensive than our RF probe based approach, we downscaled our experiments significantly, mainly:

- Experiments are run only on **Meta Llama 3.1 8B**.
- Datasets with more than **10,000 examples are subsampled** to 10,000.

This reduces the amount of data points to roughly 14% ($N = 91892$) of the original benchmarking.

J.3 Single-Source Generalization

We first evaluate in the single-source training setting where each detector is trained on one dataset and evaluated across all others. Figure 16 shows the full cross domain results.

Across all methods, we observe strong **in-domain performance** but substantial **out-of-domain degradation**. Table 9 reports the average in-domain and out-of-domain AUROC across datasets for the three methods.

Model	In-Sample AUROC	Out-of-Sample AUROC
TSV	0.793	0.685
RF	0.776	0.672
HaMI	**0.820**	**0.728**

Table 9: Average single-source performance across datasets.

We validate reported results that both HaMI and TSV outperform the RF probe based approach. Of the three, HaMI achieves the strongest average performance both in-sample and out-sample, demonstrating the best cross-domain generalization among the three methods. However, all models show noticeable degradation when evaluated outside the training domain, suggesting that out-of-domain generalization gaps remain a shared limitation across these supervised methods.

J.4 Leave-One-Out Experiments with GE Data

Next, we repeat the leave-one-out experiments described in Section 4.2, where we leave out one dataset for evaluation and train the classifiers using all remaining datasets. As before we evaluate four training strategies: TS-only, Average SS, GE-only and GE+TS.

Figure 17 shows per-benchmark results, while Table 10 reports aggregated AUROC across benchmarks.

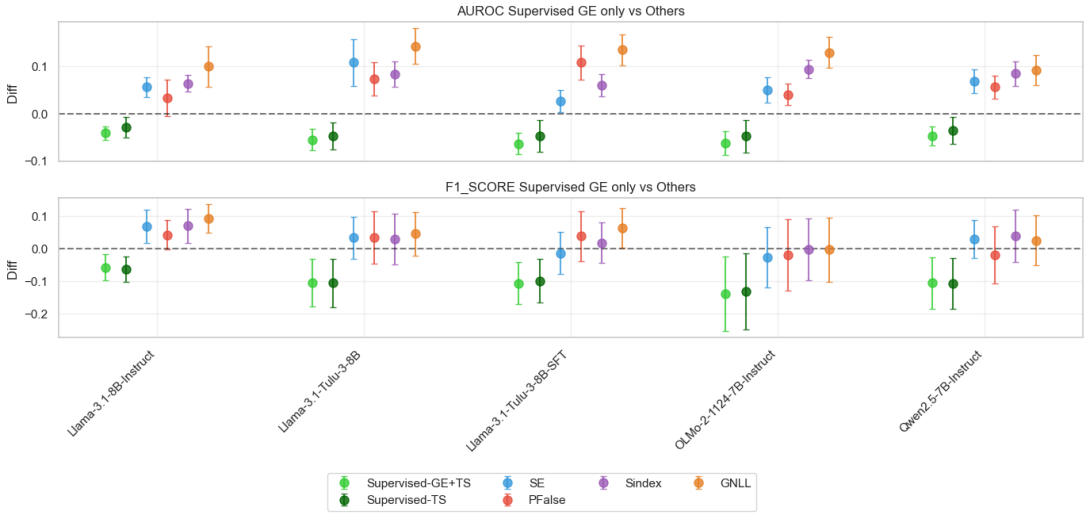
Setting	TSV	RF	HaMI
TS only	0.793	0.776	**0.820**
Average SS	0.685	0.672	**0.728**
GE	0.741	0.763	**0.796**
GE+TS	0.771	0.795	**0.820**

Table 10: Mean AUROC across benchmarks under different training strategies.

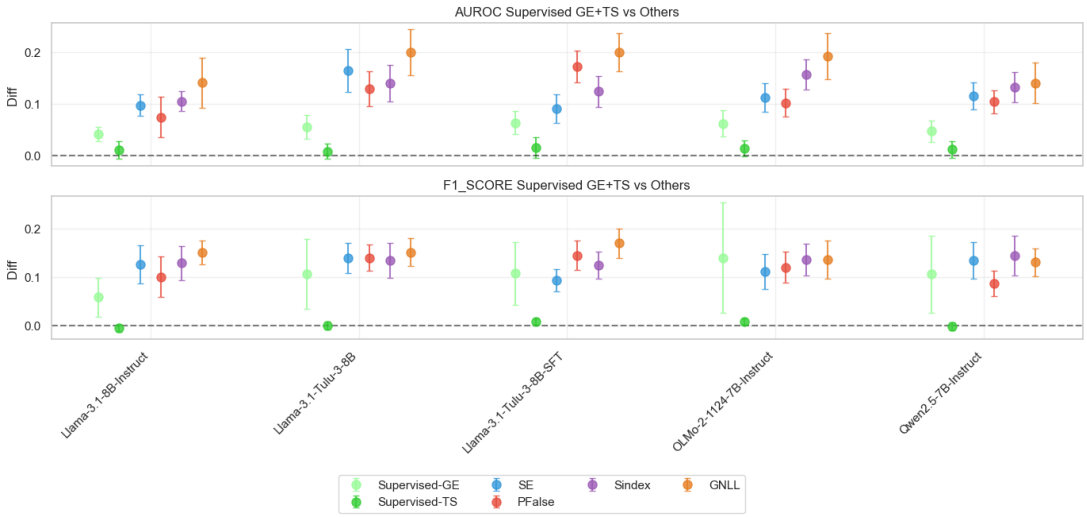
The results show that both TSV and HaMI benefit from GE training, with large improvements compared to the average single-source setting. Interestingly, TSV does not benefit from combining GE and TS data, with its the average GE+TS results underperforming TS-only. One possible explanation is that TSV learns a separating direction in activation space causing it to be less robust when trained on heterogeneous multi-domain data. In contrast, the simpler RF probe we used performs slightly better than TSV (under ths TS setting) under the GE+TS setting, suggesting that activation-based classifiers may be more robust when trained on diverse datasets. Across all settings, **HaMI remains the strongest-performing method**, achieving the highest AUROC under TS-only, GE, and GE+TS training regimes.

Importantly, the improvements observed with GE training are consistent across all three detector architectures (TSV, RF, and HaMI). This suggests that our **data-driven approach to improving hallucination detection robustness is largely model-agnostic**, and can be applied to a range of super-

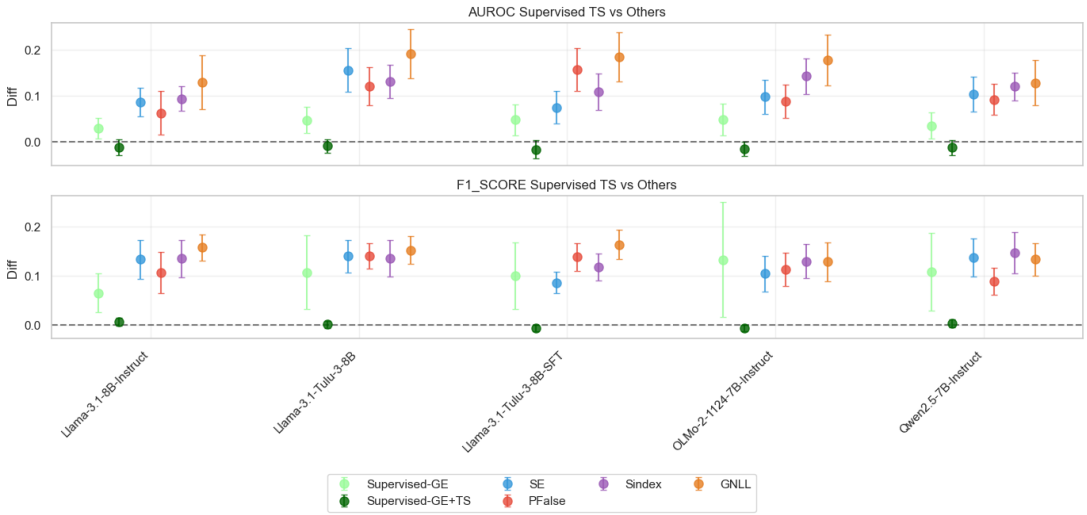
vised learning methods rather than being tied to a specific detector design.



(a) Supervised-GE vs. others



(b) Supervised-GE+TS vs. others



(c) Supervised-TS vs. others

Figure 11: Difference plots comparing the performance of (a) Supervised-GE, (b) Supervised-GE+TS, and (c) Supervised-TS against all other methods.

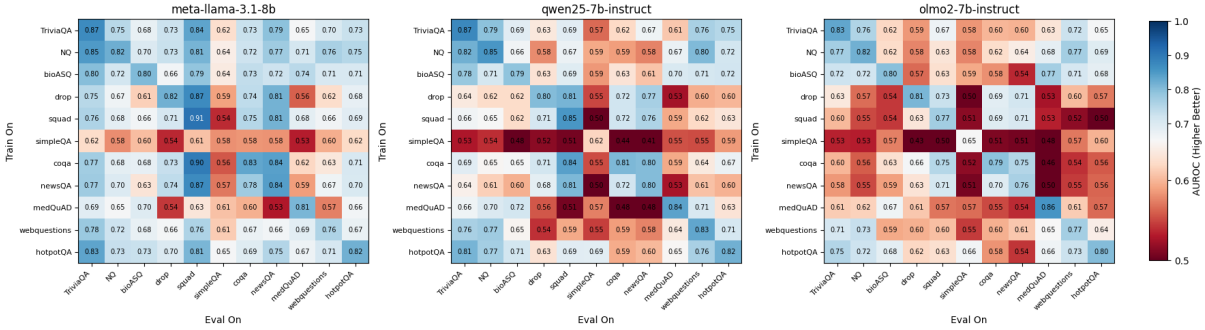


Figure 12: Single-source AUROC heatmaps across all train-eval dataset pairings for three model families. Diagonal entries represent in-domain evaluation, off-diagonal entries reflect cross-domain transfer performance.

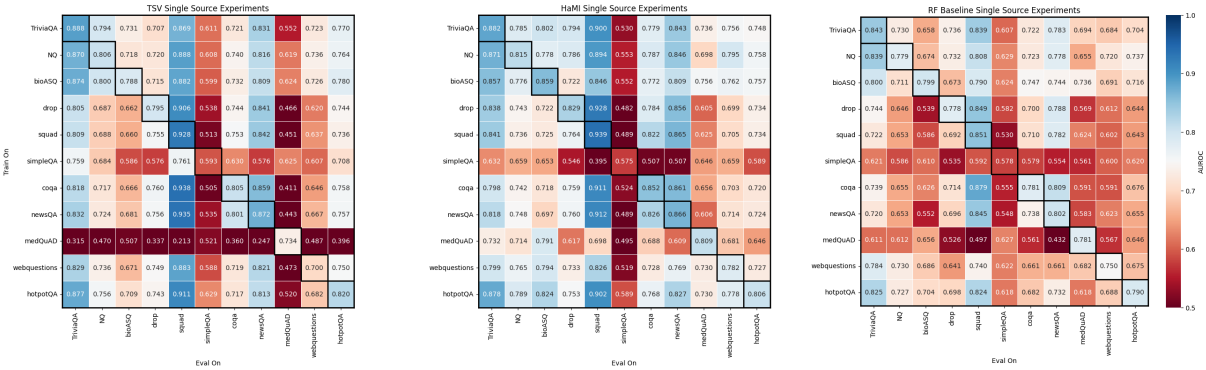


Figure 16: Single-source AUROC heatmaps across all train - evaluation dataset pairings for three supervised hallucination detection methods: TSV(Left), HaMI(Midlde), Ours (Right). Diagonal entries represent in-domain evaluation, off-diagonal entries reflect cross-domain transfer performance.

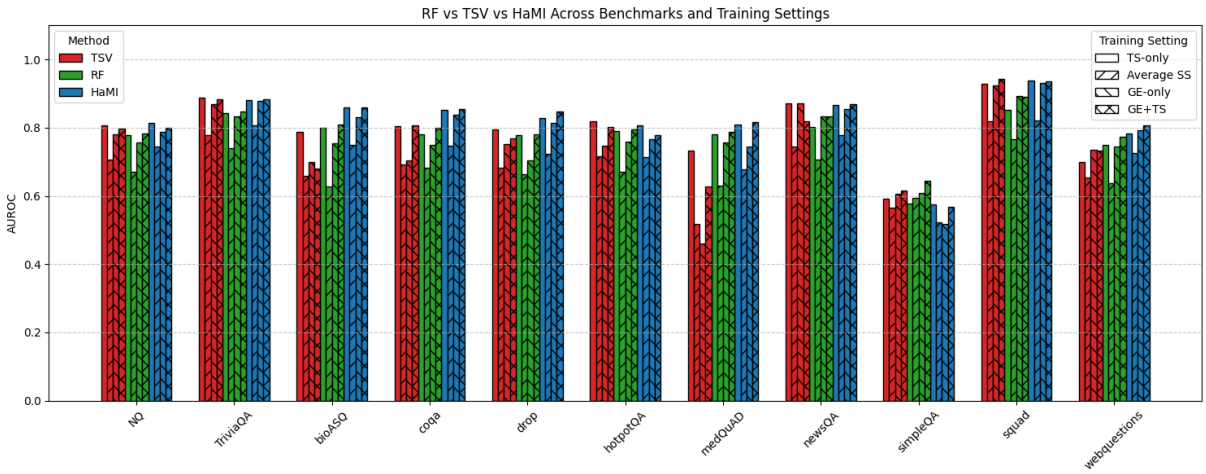


Figure 17: Disaggregated benchmark results under TS-only, Average SS, GE-Only, and GE+TS. Each color represents a supervised hallucination detection approach, TSV (red), RF Probe (Green), HaMI (Blue).

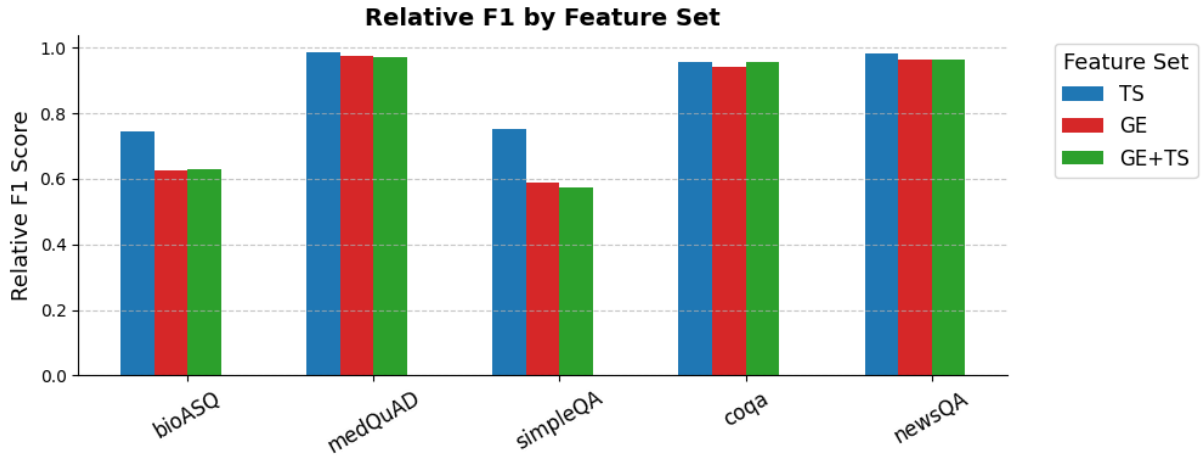


Figure 18: Relative F1-score of domain classification probes trained under 3 feature sets, TS, GE, GE+TS, scores are aggregated across 12 LLMs

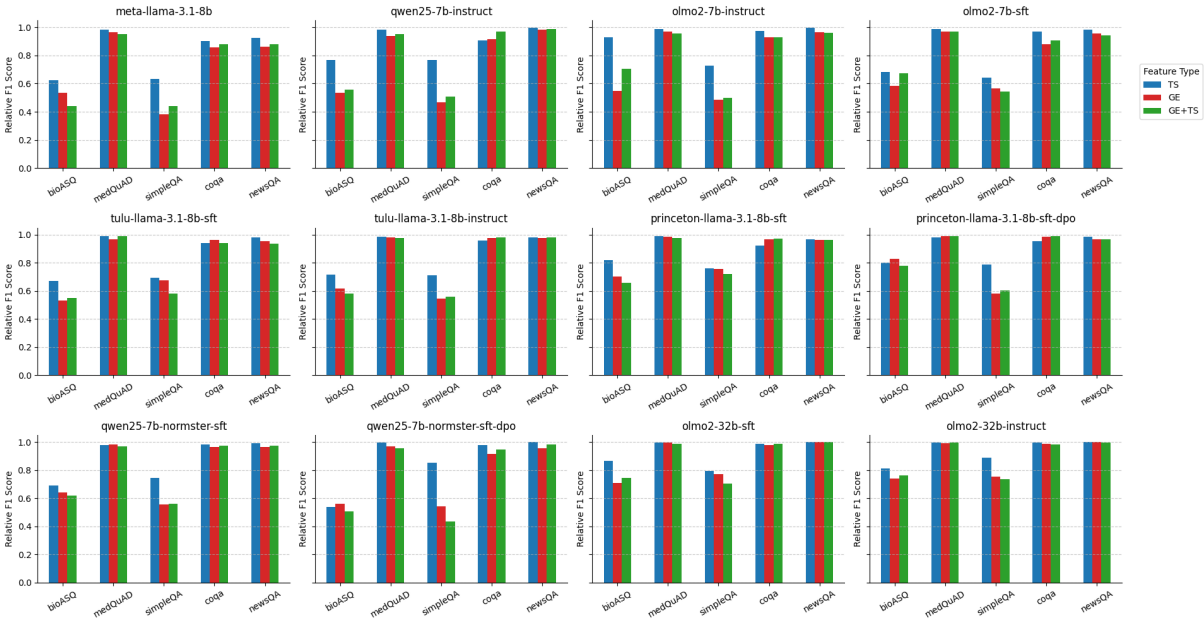


Figure 19: Relative F1-score of domain classification probes trained under 3 feature sets, TS, GE, GE+TS, each plot shows results for one of 12 tested LLMs