Unconditional Truthfulness: Learning Unconditional Uncertainty of Large Language Models

Anonymous ACL submission

Abstract

Uncertainty quantification (UQ) has emerged as a promising approach for detecting hallucinations and low-quality output of Large Language Models (LLMs). However, obtaining proper uncertainty scores is complicated by the conditional dependency between the generation steps of an autoregressive LLM, because it is hard to model it explicitly. Here, we propose to learn this dependency from attention-based features. In particular, we train a regression model that leverages LLM attention maps, probabilities on the current generation step, and recurrently computed uncertainty scores from previously generated tokens. To incorporate the recurrent features, we also suggest a two-staged training procedure. Our experimental evaluation on ten datasets and three LLMs shows that the proposed method is highly effective for selective generation, achieving substantial improvements over rivaling unsupervised and supervised approaches.

1 Introduction

Uncertainty quantification (UQ; Gal and Ghahramani (2016); Baan et al. (2023); Geng et al. (2024); Fadeeva et al. (2023)) is of growing interest in the Natural Language Processing (NLP) community for dealing with Large Language Models (LLMs) hallucinations (Fadeeva et al., 2024) and low-quality generations (Malinin and Gales, 2021) in an efficient manner. For example, high uncertainty could serve as an indicator that the LLM generation should be discarded as potentially harmful or misleading. This approach is known in the literature as selective generation (Baan et al., 2023).

There are many approaches for detecting hallucinations and low-quality outputs of LLMs (Manakul et al., 2023; Min et al., 2023; Chen et al., 2023). However, many of them leverage external knowledge sources or a second LLM. Knowledge sources are generally patchy in coverage, while censoring the outputs of a small LLM using a bigger one has a high computational cost and is impractical. We argue that LLMs inherently contain information about the limitations of their own knowledge, and that there should be an efficient way to access this information, which can enable LLM-based applications that are both safe and practical.

While for general classification and regression tasks there is a well-developed battery of UQ techniques (Zhang et al., 2019; He et al., 2020; Xin et al., 2021; Wang et al., 2022; Vazhentsev et al., 2023; He et al., 2024a), for text generation tasks, UQ is much more complicated. The complexity is multifold: (1) there is an infinite number of possible generations, which complicates the normalization of the uncertainty scores, (2) in the general case, there are an infinite number of correct answers (Farquhar et al., 2024), (3) decisions are generally based on imprecise sampling and inference algorithms such as beam search, (4) there is not one, but multiple tokens, and the uncertainty of these predictions needs to be aggregated, and finally, (5) the predictions at each generation step are not conditionally independent (Zhang et al., 2023).

This last problem is the focus of the present work. During generation, LLMs condition on the previously generated tokens. Thus, if an LLM has hallucinated and generated an incorrect claim at the beginning or in the middle of the sequence, all subsequently generated claims might also be incorrect. Even if the first claim was generated with high uncertainty, this is not taken into account during the subsequent generation process. This means that while the first error could be recognized as having high uncertainty, all subsequent errors will be overlooked because the generation process conditioned on this error will be very confident.

We note that the attention between the generated tokens provides information about the conditional dependency between the generation steps. Previously, there have been several attempts to

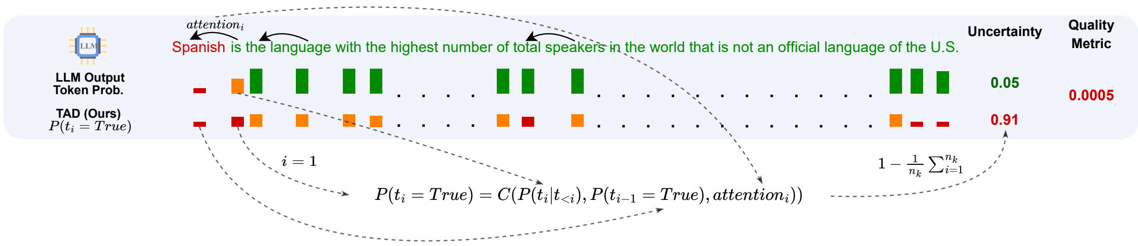

Figure 1: An illustration of the proposed method TAD. The figure shows the generated tokens, the uncertainty scores for the generated sequence, and the probabilities assigned by an LLM and by TAD (represented with bars). The output is generated by Gemma 7b for the question *What is the language with the highest number of total speakers in the world that is not an official language of the U.S.?* The LLM starts by generating a token *Spanish* that leads to the erroneous answer. The probabilities estimated by the LLM are high for all tokens except for the first one, which makes the uncertainty scores based on raw probabilities misleadingly low. On the contrary, TAD takes into account uncertainty from the previous step using a trainable model $C(\cdot)$ based on attention, resulting in a high overall uncertainty for the generated answer.

suggest heuristic approaches to model this dependency (Zhang et al., 2023). We argue that the particular algorithmic function would be too difficult to engineer, and thus we propose to learn this dependency from data instead.

For this purpose, we generate a training dataset with a target variable, representing the quality score of the generated text according to some ground truth annotation, and train a regression model that leverages LLM attention maps, probabilities on the current generation step, and recurrently computed uncertainty scores from previously generated tokens. To incorporate recurrent features, we suggest a two-staged training procedure where in the second stage, we use scores from the intermediate model obtained in the first training stage. We call the proposed approach *Trainable Attention-based Dependency* (*TAD*). Figure 1 illustrates the idea of the method on the real output of an LLM.

The **contributions** of this work are as follows.

- We develop a new data-driven supervised approach to uncertainty quantification that leverages features based on attention maps, probabilities on the current generation step, and recurrently computed uncertainty scores from previously generated tokens.
- We show that both attention and recurrent features are essential for achieving high performance in UQ, and two step training procedure is necessary to avoid overfitting.
- We conduct vast empirical investigation in selective generation and show that the proposed approach outperforms previous unsupervised and supervised UQ methods across nine datasets and three LLMs.

2 Problem Background and Key Idea

When an LLM generates a sequence of tokens t_i, it provides us a conditional probability distribution $p(t_i \mid \mathbf{t}_{<i}) = p(t_i \mid \mathbf{x}, \mathbf{t}_{<i})$, where $\mathbf{x}$ is an input prompt and $\mathbf{t}_{<i}$ is a sequence of tokens generated before token t_i. This essentially means that the LLM considers that everything generated so far is correct, which might not be the case. In practice, we would like to somehow propagate the uncertainty from the previous generation steps.

To illustrate the problem, for the sake of simplicity, let us assume that only the uncertainty from the previous tokens is propagated to the current generation step. This assumption can be expressed as follows: $p(t_i \mid \mathbf{t}_{<i}) \simeq p(t_i \mid t_{i-1})$. Let us further consider that we have trained an LLM that generates only tokens that are true ('T') or false ('F'). The probability of the token t_i being 'T' is given by the conditional probability $p(t_i \mid t_{i-1}) = p(t_i = \mathrm{T} \mid t_{i-1} = \mathrm{T})$. Assume we already have some tokens $t_1, t_2, \ldots, t_n$ and a prompt x. At each step, the LLM provides us $p(t_1 = \mathrm{T} \mid \mathbf{x}), p(t_2 = \mathrm{T} \mid t_1 = \mathrm{T}), \ldots, p(t_n = \mathrm{T} \mid t_{n-1} = \mathrm{T})$.

These probability distributions are conditionally dependent on the previously generated tokens. However, to estimate the correctness of some token t_i, we need to obtain an *unconditional probability* $p(t_i) = p(t_i = \mathrm{T})$. Let us expand $p(t_i = \mathrm{T})$ according to the law of total probability and express it using conditional probability:

$$p(t_i = \mathrm{T}) = p(t_i = \mathrm{T} \mid t_{i-1} = \mathrm{T}) \cdot p(t_{i-1} = \mathrm{T}) + p(t_i = \mathrm{T} \mid t_{i-1} = \mathrm{F}) \cdot \big(1 - p(t_{i-1} = \mathrm{T})\big).$$

In this formula, $p(t_i = \mathrm{T} \mid t_{i-1} = \mathrm{T})$ is what the LLM provides during the current generation step in accordance with the specified assumptions, and $p(t_{i-1} = \mathrm{T})$ is recurrently calculated based on the previous generation step. We still do not know the remaining term: $p(t_i = \mathrm{T} \mid t_{i-1} = \mathrm{F})$. This simplistic example shows that in order to obtain a reliable uncertainty estimate, we cannot rely solely on the probability distribution provided by the LLM, and we also need to model the conditional dependency of the generation steps. It also makes explicit the need for recurrence in token-level uncertainty computation.

Attention weights commonly reflect the degree of conditional dependency between the generation steps. However, obtaining a direct expression that would accurately approximate the conditional dependency between the generation steps is challenging. The assumptions in our simplistic example do not hold in real LLMs, and thus the predictions on each step depend on multiple previous tokens in a complicated fashion. We suggest learning this dependency in a supervised way from attention. In particular, we propose a feature set for training token-level unconditional confidence scores C, consisting of the attention weights Att_i, the token probabilities from the LLM on the current step $p(t_i \mid \mathbf{t}_{<i})$, and the recurrently calculated confidence scores on the previous steps $\boldsymbol{C}_{<i}$:

$$C(t_i) = C\big(Att_i, p(t_i \mid \mathbf{t}_{<i}), \boldsymbol{C}_{<i}\big). \quad (1)$$

3 Trainable Attention-Based Conditional Dependency

We learn unconditional token-level probability estimates and aggregate the resulting scores into a single uncertainty score for the entire sequence.

Obtaining targets for learning unconditional probability. In order to obtain the targets $\hat{p}(t_i)$ for the unconditional probability $C(t_i)$ for a generated token $t_i \in \mathbf{y}$ during the training phase, we compute the semantic similarity between the generated answer $\mathbf{y}$ and the ground truth $\mathbf{y}^*$:

$$\hat{p}(t_i) = \mathrm{sim}(\mathbf{y}, \mathbf{y}^*). \quad (2)$$

For generating the targets, we use task-specific similarity measures, such as Accuracy, COMET (Rei et al., 2020), and AlignScore (Zha et al., 2023).

Generating training data for TAD. We generate the training data for TAD using the original textual training dataset in the following way:

1. For the input prompt $\mathbf{x}_k$, we use an LLM to generate a text $\mathbf{y}_k = t_1 t_2 \dots t_{n_k}$ of some length n_k and token probabilities $p(t_i \mid \mathbf{x}_k, \mathbf{t}_{<i})$.
2. For the first generated token t_1 in each text, we introduce its unconditional confidence estimate $\hat{p}_k(t_1) = \mathrm{sim}(\mathbf{y}_k, \mathbf{y}^*_k)$ according to Equation (2).
3. For each generated token t_i, $i = 2, \dots, n_k$ we construct a feature vector z_i^k that depends on N preceding tokens. The feature vector z_i^k includes: the conditional probabilities $p(t_i \mid \mathbf{x}_k, \mathbf{t}_{<i})$ and $p(t_{i-l} \mid \mathbf{x}_k, \mathbf{t}_{<i-l})$, for $l = 1, \dots, \min\{N, i-1\}$; the unconditional probabilities' estimates from the previous steps $\hat{p}_k(t_{i-l})$, and the attention weights $a_{i,i-l}$ from the $(i-l)$-th token to the i-th token from all layers and heads. If $N > i - 1$, we pad the feature vector with zeros to ensure they have the same length. On the first stage of learning, the unconditional probabilities $\hat{p}_k(t_{i-l})$ are estimated by an auxiliary non-recursive procedure. On the subsequent learning stages it is estimated via the function learned on the previous learning stage.

As a result, for each instance in the training dataset and for each iteration of learning, we generate a sequence of target variables $\tilde{C}_i^k = \mathrm{sim}(\mathbf{y}_k, \mathbf{y}^*_k)$ and corresponding feature vectors $z_i^k, k = 1, \dots, K, i = 2, \dots, n_k$. We use this dataset to train the model C. The step-by-step procedure for generating training data is presented in Algorithm 1 in Appendix E.

Model for C and its training procedure. The training procedure involves using the estimates of the unconditional probabilities from the previous steps as features. To address this problem, we perform the training procedure twice. In the second stage, we leverage the predictions of the function C trained on the first stage as features. This two-step training approach enables us to leverage the conditional dependency of the current step on the previous ones when computing the uncertainty score. Our experiments show that it is essential for achieving good performance.

We experiment with two regression models for TAD: linear regression (LinReg) and a multi-layer perceptron (MLP). The hyper-parameters of the regressors are obtained using cross-validation with five folds on the training dataset. We select the optimal values of the hyperparameters based on the best average PRR. The optimal values are used to train the regression model on the full training set. The selected hyper-parameters values for the TAD modules are presented in Appendix C.1.

Inference procedure. During inference, we obtain predictions from the LLM as always, but we also extract features from the attention outputs. For the first generated token t_1, its unconditional probability is defined as $p(t_1) = p(t_1 \mid \mathbf{x}_k)$. For each subsequent token, the function C computes the predictions recursively, leveraging the attentions, the conditional probabilities, and the unconditional probabilities predicted for the preceding tokens. Finally, for computing uncertainty of LLM answer, the token-level scores are aggregated into a sequence-level score:

$$U(\mathbf{y}) = 1 - \frac{1}{n_k} \sum_{i=1}^{n_k} C^k(t_i). \tag{3}$$

We experiment with various aggregation approaches in the ablation study.

4 Related Work

The majority of the methods for UQ of LLM generations has been unsupervised, with few recently-proposed supervised methods.

Unsupervised UQ methods. The problem of multiple correct generations was explicitly addressed in (Kuhn et al., 2023; Nikitin et al., 2024; Cheng and Vlachos, 2024; Zhang et al., 2024) and in a series of black-box generation methods (Lin et al., 2024). The main idea is to sample multiple generations from an LLM, extract semantically equivalent clusters, and analyze the diversity of the generated meanings instead of the surface forms. Chen et al. (2024) proposed evaluating the consistency of the multiple generations in the embedding space using their hidden states. In this category, lexical similarity (Fomicheva et al., 2020) is a very competitive baseline that can be applied to black-box models (without any access to logits or internal model representations). Fadeeva et al. (2024) addressed the problem of multiple sources of uncertainty present in the LLM's probability distribution that are irrelevant for hallucination detection.

Zhang et al. (2023) and Duan et al. (2024) highlighted that not all tokens should contribute to the uncertainty score, proposing heuristics to select the relevant tokens. Zhang et al. (2023) also modeled the conditional dependencies between the generation steps by penalizing the uncertainty scores based on the uncertainties of the previously generated tokens and the max-pooled attention to the previous tokens.

Overall, most previous work on UQ has not addressed the conditional dependency between the predictions, or has addressed it using heuristics. We argue that the conditional dependency is an important aspect of UQ for text generation tasks, and we propose a data-driven approach to it. We also note that techniques based on sampling multiple answers from LLMs usually introduce prohibitive computational overhead. We argue that for UQ methods to be practical, they should also be computationally efficient.

Supervised UQ methods. Supervised regression-based confidence estimators are well-known for classification problems, primarily from computer vision (Lahlou et al., 2023; Park and Blei, 2024). Their key benefit is computational efficiency.

A handful of papers applied them to text generation tasks. Lu et al. (2022) proposed training a regression head of a model to predict confidence. They noted that the probability distribution of a language model is poorly calibrated and cannot be used directly to spot low-quality translations. They modified the model architecture and the loss function, restricting this approach to fine-tuning language models only for Machine Translation (MT) and making it unsuitable for general-purpose LLMs. In a similar vein, Azaria and Mitchell (2023) approached the task of UQ by training a multi-layer perceptron (MLP) on the activations of the internal layers of LLMs to classify true vs. false statements. They demonstrated that it outperformed other supervised baselines and few-shot prompting of the LLM itself. However, the reliance on forced decoding limits the real-world applicability for hallucination detection in unrestricted generation cases.

Several studies enhanced this method by refining the model architecture and the training procedure. Su et al. (2024) combined the hidden state of the last token with the average hidden state of the sequence, while CH-Wang et al. (2024) introduced a trainable attention layer over token embeddings and used linear regression on top of the MLP's predictions based on embeddings from various layers. He et al. (2024b) proposed to combine multiple deep learning models trained on diverse features extracted from hidden states. Chuang et al. (2024) suggested training the linear classifier using fea-

UQ Method	SamSum AlignScore	CNN AlignScore	WMT19 Comet	MedQUAD AlignScore	TruthfulQA AlignScore	CoQA AlignScore	SciQ AlignScore	TriviaQA AlignScore	MMLU Acc.	GSM8k Acc.	Mean PRR	Mean Rank
MSP	.298	.157	.569	.356	.277	.450	.582	.687	.444	.380	.420	7.30
Perplexity	.029	-.116	.460	.438	.178	.450	.202	.689	.374	.259	.296	13.10
Mean Token Entropy	.005	-.129	.444	.432	.164	.434	.199	.711	.122	.279	.266	15.00
CCP	.287	.101	.453	.321	.176	.385	.364	<u>.712</u>	.261	.408	.347	11.00
Focus	.144	-.002	.501	.460	.213	.345	.456	.621	.155	.402	.330	13.10
Simple Focus	.230	.101	.553	.381	.262	.475	.540	.703	.413	.381	.404	7.50
Lexical Similarity Rouge-L	.073	.074	.455	.153	.029	.428	.555	.613	.313	.452	.315	13.20
EigenScore	-.002	.094	.468	.047	.033	.412	.541	.591	.154	.385	.272	15.50
EVL NLI Score entail.	.111	.056	.366	.133	.134	.458	.527	.684	.304	.359	.313	13.60
Ecc. NLI Score entail.	.020	.003	.406	.099	.127	.434	.541	.632	.322	.399	.298	14.30
DegMat NLI Score entail.	.112	.062	.388	.138	.134	.453	.542	.703	.279	.385	.320	12.20
Semantic Entropy	.089	.056	.524	.027	.051	.423	.527	.660	.223	.465	.305	14.20
SAR	.121	.081	.508	.219	.078	.458	.545	.697	.299	.471	.348	9.80
LUQ	.153	.058	.258	.107	.099	.428	.499	.692	.267	.289	.285	14.70
Semantic Density	.062	.093	.347	.180	.167	.478	.497	.691	.281	.315	.311	12.60
Factoscope	.067	.086	.218	.236	.164	.049	.386	.460	.703	.108	.248	15.60
SAPLMA	.284	.073	.574	.429	.146	.039	.425	.535	.492	.508	.350	11.20
MIND	.217	.162	.494	<u>.583</u>	.385	.381	.589	.632	**.813**	.607	.486	6.90
Sheeps	.292	.179	.554	.552	<u>.464</u>	<u>.500</u>	.487	.709	.796	<u>.659</u>	.519	<u>4.40</u>
LookBackLens	**.459**	**.233**	**.615**	.579	.386	.441	<u>.594</u>	.631	.774	.619	<u>.533</u>	<u>4.40</u>
TAD	<u>.431</u>	<u>.215</u>	<u>.612</u>	**.662**	**.565**	**.509**	**.644**	**.737**	<u>.806</u>	**.682**	**.586**	**1.40**

Table 1: PRR↑ of UQ methods for the Llama-3.1 8b model. Warmer color indicates better results. The best method is in **bold**, the second best is <u>underlined</u>.

tures derived from attention matrices. A key limitation of these methods is that they can only provide veracity scores for the entire generated text.

Unlike previous methods, we focus on modeling the conditional dependencies between generation steps using attentions in a supervised way. Moreover, our method incorporates recurrently computed uncertainty scores for tokens from previous generation steps, capturing the relationship between uncertainty scores of the generated tokens. Our method is also flexible as it can be applied at different levels: to the entire text, to a sub-sequence, or to individual tokens. Finally, unlike LookBackLens, which relies on heuristically computed features, our method directly utilizes raw attention weights that give access to more information.

5 Experiments and Evaluation

5.1 Experimental Setup

For the experimental evaluation, we use the LM-Polygraph framework (Fadeeva et al., 2023). We focus on the task of selective generation (Ren et al., 2023) where we "reject" generated sequences due to low quality based on uncertainty scores. Rejecting means that we do not use the model output, and the corresponding queries are processed differently, e.g., they could be further reprocessed manually.

Evaluation measures. Following previous work on UQ in text generation (Malinin and Gales, 2021; Vashurin et al., 2025), we compare UQ methods using the Prediction Rejection Ratio (PRR) metric. PRR quantifies how well an uncertainty score can identify and reject low-quality predictions according to some quality measure. The PRR scores are normalized to the range $[0, 1]$ by linearly scaling the area under the PR curve between the values obtained with random selection (corresponding to 0) and oracle selection (corresponding to 1). Higher PRR values indicate better quality of the selective generation. We use Accuracy, COMET (Rei et al., 2020), and AlignScore (Zha et al., 2023) as generation quality measures. For QA datasets, we also use ROC-AUC of detecting incorrect answers as a supplementary metric, as it is widely adopted in the UQ literature.

Datasets. We consider ten datasets from five text generation tasks: text summarization (TS), machine translation (MT), Question Answering (QA) with long free-form answers, QA with free-form short answers, and multiple-choice QA. A detailed description of all datasets is provided in Appendix D, the dataset statistics are presented in Table 21.

LLMs. We experiment with three LLMs: LLaMA-3.1 8b (Dubey et al., 2024), Gemma-2 9b (Rivière et al., 2024), and Qwen-2.5 7b (Yang et al., 2024). The values of the inference hyper-parameters are given in Table 20 in Appendix C.2.

UQ baselines. The set of unsupervised baselines includes Maximum Sequence Probability (MSP), Mean Token Entropy, and Perplexity (Fomicheva et al., 2020), which are considered simple yet strong and robust baselines for selective generation across various tasks (Fadeeva et al., 2023). We also compare our method to unsupervised techniques considered to be state-of-the-art: Lexical Similarity based on ROUGE-L (Fomicheva

UQ Method	Llama-3.1 8b	Gemma-2 9b	Qwen-2.5 7b	Mean Rank
MSP	7.30	8.00	7.40	4.50
Perplexity	13.10	11.80	12.20	10.83
Mean Token Entropy	15.00	12.60	13.20	15.50
CCP	11.00	12.10	13.50	12.17
Focus	13.10	12.30	15.00	14.83
Simple Focus	7.50	8.20	7.50	5.67
Lexical Similarity Rouge-L	13.20	13.90	12.80	15.00
EigenScore	15.50	15.80	13.40	18.67
EVL NLI Score entail.	13.60	12.40	12.10	12.33
Ecc. NLI Score entail.	14.30	13.40	14.10	17.67
DegMat NLI Score entail.	12.20	13.20	12.00	11.17
Semantic Entropy	14.20	11.50	12.60	12.00
SAR	9.80	9.30	8.90	7.00
LUQ	14.70	13.50	13.30	17.00
Semantic Density	12.60	13.20	13.50	14.67
Factoscope	15.60	16.40	17.10	21.00
SAPLMA	11.20	10.70	13.20	10.17
MIND	6.90	7.10	7.40	3.83
Sheeps	<u>4.40</u>	9.00	6.50	3.83
LookBackLens	<u>4.40</u>	<u>5.00</u>	<u>3.50</u>	<u>2.17</u>
TAD	**1.40**	**1.60**	**1.80**	**1.00**

Table 2: Mean ranks of UQ methods aggregated over all datasets for each LLM separately (the lower the better). The column *Mean Rank* corresponds to the mean rank of the ranks across all LLMs. The best method is in **bold**, the second best is <u>underlined</u>.

et al., 2020), black-box methods (DegMat, Eccentricity, EigValLaplacian; Lin et al. (2024)), Semantic Entropy (Kuhn et al., 2023), hallucination detection with a stronger focus (Focus; Zhang et al. (2023)), claim-conditioned probability (CCP; Fadeeva et al. (2024)), Shifting Attention to Relevance (SAR; Duan et al. (2024)), EigenScore (Chen et al., 2024), Semantic Density (Qiu and Miikkulainen, 2024), and long-text uncertainty quantification (LUQ; Zhang et al. (2024)). For sampling-based methods, we generate five samples.

The suite of baselines also includes state-of-the-art supervised methods that use hidden states or attention weights: Factoscope (He et al., 2024b), SAPLMA (Azaria and Mitchell, 2023), MIND (Su et al., 2024), Sheeps (CH-Wang et al., 2024), and LookBackLens (Chuang et al., 2024).

5.2 Main Results

Fine-grained comparison to the baselines. Tables 1, 5 and 6 in Appendix A.1 present the results for LLaMa-3.1 8b, Gemma-2 9b, and Qwen-2.5 7b, respectively.

The results demonstrate that, across all summarization and translation datasets, both LookBackLens and TAD outperform state-of-the-art methods by a substantial margin. For Llama, LookBackLens achieves slightly better results than TAD, but TAD confidently outperforms LookBackLens on the CNN dataset when using Gemma and on the WMT19 dataset with Qwen.

For QA involving long answers (e.g., MedQUAD, TruthfulQA, and GSM8k), TAD demonstrates substantial improvements over the baselines across all considered models. For example, in the experiment with LLaMA-3.1 8b on TruthfulQA, TAD outperforms the second-best baseline, Sheeps, by 0.101 of PRR. On the MedQUAD dataset, TAD achieves an improvement of 0.079 in PRR over the second-best baseline, and on GSM8k, it improves PRR by 0.023.

For QA with short answers (CoQA, SciQ, and TriviaQA), TAD generally exhibits notable improvements over the baseline methods in the majority of cases. The only exception is the case of the SciQ dataset, where LookBackLens is marginally better for Gemma-2 9b and Qwen-2.5 7b. On TriviaQA, when using the Gemma-2 9b model, TAD performs on par with sampling-based methods, while other supervised methods fall behind simple baselines by a margin.

Finally, for MMLU, TAD also notably outperforms state-of-the-art methods for both Gemma-2 9b and Qwen-2.5 7b. However, for LLaMA-3.1 8b, TAD slightly falls behind MIND.

Summarizing, our findings indicate that certain UQ methods, such as LookBackLens and Sheeps, can achieve top performance in specific experimental settings. However, TAD demonstrates the most consistent and robust performance across all eleven tasks, never ranking below the second-best method. In contrast, other supervised methods occasionally underperform, sometimes even falling below simple baselines such as MSP. Similar patterns are observed in the ROC AUC results reported in Tables 7 to 9 (see Appendix A.2).

Aggregated results. Table 2 presents the mean rank of each method aggregated over all datasets for each model separately. The lower rank is better. The column *Mean Rank* shows the mean rank of the ranks across all models. Figure 2 additionally summarizes all experimental setups. Each cell presents a win rate for a method from a row compared to a method from a column. The aggregated results emphasize the significance of the performance improvements of the proposed method. Despite some baselines showing good results in particular cases, they usually are quite unstable, resulting in poor overall ranking. TAD demonstrates more robust improvements across multiple tasks and LLMs, making it a better choice overall.

Generalization to out-of-domain datasets. Table 3 compares the results of the supervised methods trained on all QA datasets except for one that represents the out-of-domain dataset for testing. Additionally, Table 10 in Appendix A.3 presents

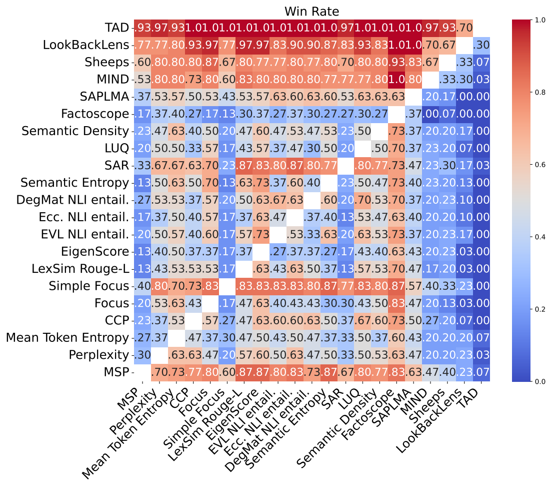

Figure 2: Summary of 30 experimental setups with various models and datasets. Each cell in the diagram presents the fraction of experiments where a method from a row outperforms a method from a column. Warmer colors indicate better results.

UQ Method	MedQUAD AlignScore	CoQA AlignScore	SciQ AlignScore	MMLU Acc.	GSM8k Acc.	Mean PRR
MSP	**.356**	<u>.450</u>	<u>.582</u>	.444	<u>.380</u>	<u>.442</u>
Factoscope	.166	.007	.129	-.022	-.082	.039
SAPLMA	.137	.012	.270	-.034	.073	.092
MIND	.095	.171	-.045	.415	.335	.194
Sheeps	.044	.201	.538	**.624**	.348	.351
LookBackLens	.061	.111	.407	.224	.261	.213
TAD	<u>.336</u>	**.461**	**.629**	<u>.489</u>	**.391**	**.461**

Table 3: PRR↑ for Llama 8b v3.1 model for various QA tasks for the considered supervised sequence-level methods trained on the general QA dataset. Unsupervised methods are not included as their performance is not dependent of the training data. Warmer colors indicate better results. The best method is in **bold**, and the second best one is <u>underlined</u>.

the results when these methods are trained on all QA datasets and tested on the out-of-distribution tasks: summarization and translation. These settings evaluate the out-of-domain generalization capabilities of the supervised techniques for both new domains and new tasks.

The results show that all considered supervised methods substantially degrade compared to their in-domain performance and, in many cases, underperform the simple MSP baseline. Nevertheless, TAD demonstrates strong out-of-domain performance on the unseen QA datasets, outperforming MSP by 0.019 of PRR on average. However, all supervised methods perform significantly worse than the MSP baseline on the OOD tasks, summarization and translation, underscoring their limited adaptability to unseen tasks.

These findings indicate that previous supervised UQ methods are generally effective only for in-domain selective generation. However, the TAD method demonstrates the ability to achieve generalization to unseen domains within similar tasks. More details about these experiments are presented in Appendix A.3.

5.3 Ablation Studies

Comparison of features. Table 15 in Appendix A.5 presents the ablation experiment with different features for the TAD regression model. For *TAD (probs.)*, we only use probabilities along with predictions from the preceding tokens $p(t_{i-k} = \text{T})$ for $k = 1, \dots, N$. For *TAD (attention)*, we use attention weights on the N preceding tokens without probabilities. The results show that *TAD (probs.)* provides meaningful but relatively low performance. *TAD (attention)* demonstrates substantial improvements, underscoring the importance of using the attentions in the TAD method. Finally, *TAD (attention+probs.)*, which combines both attention weights, probabilities, and uncertainty scores from previous steps, achieves slight but consistent performance gains. This indicates the benefit of recurrence during the computation of uncertainty scores.

Impact of the token-level training procedure. Table 14 in Appendix A.5 presents an ablation study comparing different training procedures for the regression model in the TAD method. We compare the original TAD against *TAD (Sequence-level)*, which uses a two-layer MLP with averaging of the hidden features between layers, followed by a linear layer for direct sequence-level uncertainty prediction. The results demonstrate that while *TAD (Sequence-level)* performs competitively, the original TAD method surpasses it by 0.023 of PRR on average, with the largest improvement of 0.078 PRR on MedQUAD. These findings highlight the effectiveness of the token-level training procedure with recurrent features in TAD.

Impact of the two-step training procedure. Table 16 in Appendix A.5 presents the ablation experiment comparing one-step vs. two-step training procedures for the TAD method. The results show that the two-step procedure is essential for training a well-performing recurrent model.

Regression models and aggregation approaches. Detailed results with various regression models and aggregation approaches are presented in Table 12. The optimal values of the hyper-parameters of TAD for all experimental setups are presented in Tables 17 to 19 in Appendix C.1 for LLaMA-3.1 8b, Gemma-2 9b, and Qwen-2.5 7b, respectively.

We compared two strategies for aggregating the token-level TAD scores: (*i*) the mean of the scores and (*ii*) the sum of the log scores inspired by perplexity. For the majority of the considered settings, the mean of the probabilities yielded the best results. However, for QA with short answers, the sum of the log probabilities performed slightly better.

We can see that the difference between MLP and LinReg is minimal. On average, TAD with LinReg outperforms TAD with MLP by 0.029 in PRR. Therefore, for simplicity, we use LinReg as a regression method for TAD.

Impact of the number of previous tokens. Table 13 presents experiments with different numbers of preceding tokens used in TAD. The results show that using ten preceding tokens generally yields better performance compared to using only 1-2 tokens across all datasets, except for SamSum.

Impact of the attention layers. Figure 3 in Appendix A.5 presents the normalized average weights of linear regression for different attention layers in the TAD method. We can see similar patterns across various tasks, revealing that the most important layers are typically the middle ones, which is consistent with observations in previous work (Azaria and Mitchell, 2023; Chen et al., 2024). Additionally, we note that for the majority of the tasks, the first and the last attention layers play a crucial role.

Replacing attention weights with interpretability features. Table 11 in Appendix A.4 shows the results, where we investigate interpretability features from Layer Integrated Gradients (LIG; Sundararajan et al. (2017)) as a measure of conditional dependency between generation steps. We compare the original TAD method with two variants: *TAD (LIG)*, which replaces attention weights with LIG features, and *TAD (MIX)*, which concatenates LIG features with the raw attention weights. LIG features perform comparably to attention, but their inclusion does not enhance TAD performance.

5.4 Computational Efficiency

In order to demonstrate the computational efficiency of TAD, we compare its runtime to other UQ methods. We use a single 80GB H100 GPU, as detailed in Table 1. The inference is implemented as a single-batch model call for all tokens in the output text.

Table 4 presents the average runtime per text instance for each UQ method, along with the percentage overhead over the standard LLM inference with MSP. As we can see, many state-of-the-art UQ methods such as (DegMat, Lexical Similarity, Semantic Entropy, and SAR) introduce huge computational overhead (400-600%) because they need to perform sampling from the LLM multiple times. In contrast, all supervised methods introduce minimal overhead. In particular, TAD introduces only 5% overhead, which makes it a highly practical and efficient choice for uncertainty quantification.

UQ Method	**Runtime per batch**	**Overhead**
MSP	1.30 ± 0.62	-
DegMat NLI Score Entail.	6.86 ± 2.28	430 %
Lexical Similarity ROUGE-L	6.72 ± 2.24	420%
Semantic Entropy	6.86 ± 2.28	430%
SAR	8.83 ± 2.94	580%
Factoscope	3.30 ± 2.13	150%
SAPLMA	1.30 ± 0.62	**0.06%**
MIND	1.30 ± 0.62	0.10%
Sheeps	1.50 ± 0.97	15%
LookBackLens	1.30 ± 0.62	<u>0.08%</u>
TAD	1.37 ± 0.68	5%

Table 4: Evaluation of the inference runtime of UQ methods measured on all test instances from all datasets with predictions from Llama 8b v3.1. The best results are in **bold**, and the second best results are <u>underlined</u>.

6 Conclusion and Future Work

We have presented a new uncertainty quantification method based on learning conditional dependencies between the predictions made on multiple generation steps. The method relies on attention to construct features for learning this functional dependency and leverages this dependency to alter the uncertainty of the subsequent generation steps. This yields improved results in selective generation tasks, especially when the LLM output is long. Our experimental study shows that TAD usually outperforms other state-of-the-art UQ methods (such as SAR) resulting in the best overall performance across three LLMs and nine datasets. Contrary to other supervised methods, TAD also shows cross-domain generalization. Our method requires only minimal computational overhead due to the simplicity of the underlying linear regression model, making it a practical choice for LLM-based applications.

In future work, we aim to apply the suggested method to UQ of retrieval-augmented LLMs. TAD potentially could be used to take into account the credibility of the retrieved evidence.

Limitations

The proposed approach is supervised and thus benefits from task-specific training data. We evaluate our method on out-of-domain data to explore its generalization. Despite expected variations in performance, the proposed method achieves promising results on unseen out-of-domain data when trained on the related source domain. Overall, the method can be used in out-of-domain settings, while caution should be exercised when training on significantly different domains.

Our experiments were conducted using 7–9B parameter models, due to limitations in our available computational resources. Nevertheless, given the similar architectures and training procedures across model scales, we believe that the proposed method can be effectively applied to larger-scale LLMs.

Ethical Considerations

In our work, we considered open-weights LLMs and datasets not aimed at harmful content. However, LLMs may generate potentially damaging texts for various groups of people. Uncertainty quantification techniques can help create more reliable use of neural networks. Moreover, they can be applied to detecting harmful generation, but this is not our intention.

Moreover, despite that our proposed method demonstrates sizable performance improvements, it can still mistakenly highlight correct and not dangerous generated text with high uncertainty in some cases. Thus, as with other uncertainty quantification methods, it has limited applicability.

References

Asma Ben Abacha and Dina Demner-Fushman. 2019. A question-entailment approach to question answering. *BMC Bioinform.*, 20(1):511:1–511:23.

Amos Azaria and Tom Mitchell. 2023. The internal state of an LLM knows when it's lying. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 967–976, Singapore. Association for Computational Linguistics.

Joris Baan, Nico Daheim, Evgenia Ilia, Dennis Ulmer, Haau-Sing Li, Raquel Fernández, Barbara Plank, Rico Sennrich, Chrysoula Zerva, and Wilker Aziz. 2023. Uncertainty in natural language generation: From theory to applications. *arXiv preprint arXiv:2307.15703*.

Loïc Barrault, Ondřej Bojar, Marta R. Costa-jussà, Christian Federmann, Mark Fishel, Yvette Graham, Barry Haddow, Matthias Huck, Philipp Koehn, Shervin Malmasi, Christof Monz, Mathias Müller, Santanu Pal, Matt Post, and Marcos Zampieri. 2019. Findings of the 2019 Conference on Machine Translation (WMT19). In *Proceedings of the Fourth Conference on Machine Translation (Volume 2: Shared Task Papers, Day 1)*, pages 1–61, Florence, Italy. Association for Computational Linguistics.

Sky CH-Wang, Benjamin Van Durme, Jason Eisner, and Chris Kedzie. 2024. Do androids know they're only dreaming of electric sheep? In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 4401–4420, Bangkok, Thailand. Association for Computational Linguistics.

Chao Chen, Kai Liu, Ze Chen, Yi Gu, Yue Wu, Mingyuan Tao, Zhihang Fu, and Jieping Ye. 2024. INSIDE: llms' internal states retain the power of hallucination detection. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net.

Yuyan Chen, Qiang Fu, Yichen Yuan, Zhihao Wen, Ge Fan, Dayiheng Liu, Dongmei Zhang, Zhixu Li, and Yanghua Xiao. 2023. Hallucination detection: Robustly discerning reliable answers in large language models. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management*, pages 245–255.

Julius Cheng and Andreas Vlachos. 2024. Measuring uncertainty in neural machine translation with similarity-sensitive entropy. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2115–2128, St. Julian's, Malta. Association for Computational Linguistics.

Yung-Sung Chuang, Linlu Qiu, Cheng-Yu Hsieh, Ranjay Krishna, Yoon Kim, and James R. Glass. 2024. Lookback lens: Detecting and mitigating contextual hallucinations in large language models using only attention maps. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 1419–1436, Miami, Florida, USA. Association for Computational Linguistics.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, Christopher Hesse, and John Schulman. 2021. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*.

Jinhao Duan, Hao Cheng, Shiqi Wang, Alex Zavalny, Chenan Wang, Renjing Xu, Bhavya Kailkhura, and Kaidi Xu. 2024. Shifting attention to relevance: Towards the predictive uncertainty quantification of free-form large language models. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5050–5063, Bangkok, Thailand. Association for Computational Linguistics.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, Arun Rao, Aston Zhang, Aurélien Rodriguez, Austen Gregerson, Ava Spataru, Baptiste Rozière, Bethany Biron, Binh Tang, Bobbie Chern, Charlotte Caucheteux, Chaya Nayak, Chloe Bi, Chris Marra, Chris McConnell, Christian Keller, Christophe Touret, Chunyang Wu, Corinne Wong, Cristian Canton Ferrer, Cyrus Nikolaidis, Damien Allonsius, Daniel Song, Danielle Pintz, Danny Livshits, David Esiobu, Dhruv Choudhary, Dhruv Mahajan, Diego Garcia-Olano, Diego Perino, Dieuwke Hupkes, Egor Lakomkin, Ehab AlBadawy, Elina Lobanova, Emily Dinan, Eric Michael Smith, Filip Radenovic, Frank Zhang, Gabriel Synnaeve, Gabrielle Lee, Georgia Lewis Anderson, Graeme Nail, Grégoire Mialon, Guan Pang, Guillem Cucurell, Hailey Nguyen, Hannah Korevaar, Hu Xu, Hugo Touvron, Iliyan Zarov, Imanol Arrieta Ibarra, Isabel M. Kloumann, Ishan Misra, Ivan Evtimov, Jade Copet, Jaewon Lee, Jan Geffert, Jana Vranes, Jason Park, Jay Mahadeokar, Jeet Shah, Jelmer van der Linde, Jennifer Billock, Jenny Hong, Jenya Lee, Jeremy Fu, Jianfeng Chi, Jianyu Huang, Jiawen Liu, Jie Wang, Jiecao Yu, Joanna Bitton, Joe Spisak, Jongsoo Park, Joseph Rocca, Joshua Johnstun, Joshua Saxe, Junteng Jia, Kalyan Vasuden Alwala, Kartikeya Upasani, Kate Plawiak, Ke Li, Kenneth Heafield, Kevin Stone, and et al. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

Ekaterina Fadeeva, Aleksandr Rubashevskii, Artem Shelmanov, Sergey Petrakov, Haonan Li, Hamdy Mubarak, Evgenii Tsymbalov, Gleb Kuzmin, Alexander Panchenko, Timothy Baldwin, Preslav Nakov, and Maxim Panov. 2024. Fact-checking the output of large language models via token-level uncertainty quantification. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 9367–9385, Bangkok, Thailand. Association for Computational Linguistics.

Ekaterina Fadeeva, Roman Vashurin, Akim Tsvigun, Artem Vazhentsev, Sergey Petrakov, Kirill Fedyanin, Daniil Vasilev, Elizaveta Goncharova, Alexander Panchenko, Maxim Panov, Timothy Baldwin, and Artem Shelmanov. 2023. LM-polygraph: Uncertainty estimation for language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 446–461, Singapore. Association for Computational Linguistics.

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

Marina Fomicheva, Shuo Sun, Lisa Yankovskaya, Frédéric Blain, Francisco Guzmán, Mark Fishel, Nikolaos Aletras, Vishrav Chaudhary, and Lucia Specia. 2020. Unsupervised quality estimation for neural machine translation. *Transactions of the Association for Computational Linguistics*, 8:539–555.

Yarin Gal and Zoubin Ghahramani. 2016. Dropout as a Bayesian approximation: Representing model uncertainty in deep learning. In *Proceedings of The 33rd International Conference on Machine Learning*, volume 48 of *Proceedings of Machine Learning Research*, pages 1050–1059, New York, New York, USA. PMLR.

Jiahui Geng, Fengyu Cai, Yuxia Wang, Heinz Koeppl, Preslav Nakov, and Iryna Gurevych. 2024. A survey of confidence estimation and calibration in large language models. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 6577–6595, Mexico City, Mexico. Association for Computational Linguistics.

Bogdan Gliwa, Iwona Mochol, Maciej Biesek, and Aleksander Wawer. 2019. SAMSum corpus: A human-annotated dialogue dataset for abstractive summarization. In *Proceedings of the 2nd Workshop on New Frontiers in Summarization*, pages 70–79, Hong Kong, China. Association for Computational Linguistics.

Jianfeng He, Linlin Yu, Shuo Lei, Chang-Tien Lu, and Feng Chen. 2024a. Uncertainty estimation on sequential labeling via uncertainty transmission. In *Findings of the Association for Computational Linguistics: NAACL 2024*, pages 2823–2835, Mexico City, Mexico. Association for Computational Linguistics.

Jianfeng He, Xuchao Zhang, Shuo Lei, Zhiqian Chen, Fanglan Chen, Abdulaziz Alhamadani, Bei Xiao, and ChangTien Lu. 2020. Towards more accurate uncertainty estimation in text classification. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 8362–8372, Online. Association for Computational Linguistics.

Jinwen He, Yujia Gong, Zijin Lin, Cheng'an Wei, Yue Zhao, and Kai Chen. 2024b. LLM factoscope: Uncovering LLMs' factual discernment through measuring inner states. In *Findings of the Association for Computational Linguistics ACL 2024*, pages 10218–10230, Bangkok, Thailand and virtual meeting. Association for Computational Linguistics.

Dan Hendrycks, Collin Burns, Steven Basart, Andy Zou, Mantas Mazeika, Dawn Song, and Jacob Steinhardt. 2021. Measuring massive multitask language understanding. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net.

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

Narine Kokhlikyan, Vivek Miglani, Miguel Martin, Edward Wang, Bilal Alsallakh, Jonathan Reynolds, Alexander Melnikov, Natalia Kliushkina, Carlos Araya, Siqi Yan, and Orion Reblitz-Richardson. 2020. Captum: A unified and generic model interpretability library for pytorch. *Preprint*, arXiv:2009.07896.

Lorenz Kuhn, Yarin Gal, and Sebastian Farquhar. 2023. Semantic uncertainty: Linguistic invariances for uncertainty estimation in natural language generation. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*. OpenReview.net.

Salem Lahlou, Moksh Jain, Hadi Nekoei, Victor I Butoi, Paul Bertin, Jarrid Rector-Brooks, Maksym Korablyov, and Yoshua Bengio. 2023. DEUP: Direct epistemic uncertainty prediction. *Transactions on Machine Learning Research*.

Stephanie Lin, Jacob Hilton, and Owain Evans. 2022. TruthfulQA: Measuring how models mimic human falsehoods. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3214–3252, Dublin, Ireland. Association for Computational Linguistics.

Zhen Lin, Shubhendu Trivedi, and Jimeng Sun. 2024. Generating with confidence: Uncertainty quantification for black-box large language models. *Transactions on Machine Learning Research*.

Yu Lu, Jiali Zeng, Jiajun Zhang, Shuangzhi Wu, and Mu Li. 2022. Learning confidence for transformer-based neural machine translation. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2353–2364, Dublin, Ireland. Association for Computational Linguistics.

Andrey Malinin and Mark J. F. Gales. 2021. Uncertainty estimation in autoregressive structured prediction. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net.

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

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

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.

Yookoon Park and David Blei. 2024. Density uncertainty layers for reliable uncertainty estimation. In *International Conference on Artificial Intelligence and Statistics*, pages 163–171. PMLR.

Xin Qiu and Risto Miikkulainen. 2024. Semantic density: Uncertainty quantification for large language models through confidence measurement in semantic space. In *Advances in Neural Information Processing Systems*, volume 37, pages 134507–134533. Curran Associates, Inc.

Siva Reddy, Danqi Chen, and Christopher D. Manning. 2019. CoQA: A conversational question answering challenge. *Transactions of the Association for Computational Linguistics*, 7:249–266.

Ricardo Rei, Craig Stewart, Ana C Farinha, and Alon Lavie. 2020. COMET: A neural framework for MT evaluation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2685–2702, Online. Association for Computational Linguistics.

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

Morgane Rivière, Shreya Pathak, Pier Giuseppe Sessa, Cassidy Hardin, Surya Bhupatiraju, Léonard Hussenot, Thomas Mesnard, Bobak Shahriari, Alexandre Ramé, Johan Ferret, Peter Liu, Pouya Tafti, Abe Friesen, Michelle Casbon, Sabela Ramos, Ravin Kumar, Charline Le Lan, Sammy Jerome, Anton Tsitsulin, Nino Vieillard, Piotr Stanczyk, Sertan Girgin, Nikola Momchev, Matt Hoffman, Shantanu Thakoor, Jean-Bastien Grill, Behnam Neyshabur, Olivier Bachem, Alanna Walton, Aliaksei Severyn, Alicia Parrish, Aliya Ahmad, Allen Hutchison, Alvin Abdagic, Amanda Carl, Amy Shen, Andy Brock, Andy Coenen, Anthony Laforge, Antonia Paterson, Ben Bastian, Bilal Piot, Bo Wu, Brandon Royal, Charlie Chen, Chintu Kumar, Chris Perry, Chris Welty, Christopher A. Choquette-Choo, Danila Sinopalnikov, David Weinberger, Dimple Vijaykumar, Dominika Rogozinska, Dustin Herbison, Elisa Bandy, Emma Wang, Eric Noland, Erica Moreira, Evan Senter, Evgenii Eltyshev, Francesco Visin, Gabriel Rasskin, Gary Wei, Glenn Cameron, Gus Martins, Hadi Hashemi, Hanna Klimczak-Plucinska, Harleen Batra, Harsh Dhand, Ivan Nardini, Jacinda Mein, Jack Zhou, James Svensson, Jeff Stanway, Jetha Chan, Jin Peng Zhou, Joana Carrasqueira, Joana Iljazi, Jocelyn Becker, Joe Fernandez, Joost van Amersfoort, Josh Gordon, Josh Lipschultz, Josh Newlan, Ju-yeong Ji, Kareem Mohamed, Kartikeya Badola, Kat Black, Katie Millican, Keelin McDonell, Kelvin Nguyen, Kiranbir Sodhia, Kish Greene, Lars Lowe Sjösund, Lauren Usui, Laurent Sifre, Lena Heuermann, Leticia Lago, and Lilly McNealus. 2024. Gemma 2: Improving open language models at a practical size. *arXiv preprint arXiv:2408.00118*.

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

Weihang Su, Changyue Wang, Qingyao Ai, Yiran Hu, Zhijing Wu, Yujia Zhou, and Yiqun Liu. 2024. Unsupervised real-time hallucination detection based on the internal states of large language models. In *Findings of the Association for Computational Linguistics: ACL 2024*, pages 14379–14391, Bangkok, Thailand. Association for Computational Linguistics.

Mukund Sundararajan, Ankur Taly, and Qiqi Yan. 2017. Axiomatic attribution for deep networks. In *Proceedings of the 34th International Conference on Machine Learning, ICML 2017, Sydney, NSW, Australia, 6-11 August 2017*, volume 70 of *Proceedings of Machine Learning Research*, pages 3319–3328. PMLR.

Roman Vashurin, Ekaterina Fadeeva, Artem Vazhentsev, Lyudmila Rvanova, Daniil Vasilev, Akim Tsvigun, Sergey Petrakov, Rui Xing, Abdelrahman Sadallah, Kirill Grishchenkov, et al. 2025. Benchmarking uncertainty quantification methods for large language models with lm-polygraph. *Transactions of the Association for Computational Linguistics*, 13:220–248.

Artem Vazhentsev, Gleb Kuzmin, Akim Tsvigun, Alexander Panchenko, Maxim Panov, Mikhail Burtsev, and Artem Shelmanov. 2023. Hybrid uncertainty quantification for selective text classification in ambiguous tasks. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11659–11681, Toronto, Canada. Association for Computational Linguistics.

Yuxia Wang, Daniel Beck, Timothy Baldwin, and Karin Verspoor. 2022. Uncertainty estimation and reduction of pre-trained models for text regression. *Transactions of the Association for Computational Linguistics*, 10:680–696.

Johannes Welbl, Nelson F. Liu, and Matt Gardner. 2017. Crowdsourcing multiple choice science questions. In *Proceedings of the 3rd Workshop on Noisy User-generated Text*, pages 94–106, Copenhagen, Denmark. Association for Computational Linguistics.

Ji Xin, Raphael Tang, Yaoliang Yu, and Jimmy Lin. 2021. The art of abstention: Selective prediction and error regularization for natural language processing. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 1040–1051, Online. Association for Computational Linguistics.

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, Tingyu Xia, Xingzhang Ren, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yu Wan, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zihan Qiu. 2024. Qwen2.5 technical report. *arXiv preprint arXiv:2412.15115*.

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

Caiqi Zhang, Fangyu Liu, Marco Basaldella, and Nigel Collier. 2024. LUQ: Long-text uncertainty quantification for LLMs. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 5244–5262, Miami, Florida, USA. Association for Computational Linguistics.

Tianhang Zhang, Lin Qiu, Qipeng Guo, Cheng Deng, Yue Zhang, Zheng Zhang, Chenghu Zhou, Xinbing Wang, and Luoyi Fu. 2023. Enhancing uncertainty-based hallucination detection with stronger focus. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 915–932, Singapore. Association for Computational Linguistics.

Xuchao Zhang, Fanglan Chen, Chang-Tien Lu, and Naren Ramakrishnan. 2019. Mitigating uncertainty in document classification. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 3126–3136, Minneapolis, Minnesota. Association for Computational Linguistics.

A Additional Experimental Results

A.1 Comparison with other UQ Methods

Here, we present the main results for Gemma and Qwen.

UQ Method	SamSum AlignScore	CNN AlignScore	WMT19 Comet	MedQUAD AlignScore	TruthfulQA AlignScore	CoQA AlignScore	SciQ AlignScore	TriviaQA AlignScore	MMLU Acc.	GSM8k Acc.	Mean PRR	Mean Rank
MSP	.370	.061	.588	.125	.187	.527	.614	.772	.771	.425	.444	8.00
Perplexity	.008	-.036	.480	.354	.171	.517	.178	.779	.756	.225	.343	11.80
Mean Token Entropy	-.039	-.066	.443	.345	.141	.475	.191	**.792**	.759	.275	.332	12.60
CCP	.266	.031	.432	.306	.102	.448	.450	.769	.678	.482	.396	12.10
Focus	.110	-.040	.494	.200	.198	.446	.528	.721	.721	.419	.380	12.30
Simple Focus	.308	.066	.578	.156	.178	<u>.543</u>	.583	.770	.755	.436	.437	8.20
Lexical Similarity Rouge-L	.077	.071	.458	.011	-.002	.453	.453	.751	.587	.544	.340	13.90
EigenScore	.134	.085	.368	.141	-.144	.456	.452	.701	.473	.355	.302	15.80
EVL NLI Score entail.	.143	.089	.373	.189	.035	.469	.464	.750	.606	.486	.361	12.40
Ecc. NLI Score entail.	.073	.047	.393	.209	-.020	.487	.478	.742	.609	.512	.353	13.40
DegMat NLI Score entail.	.147	.090	.381	.132	.034	.427	.466	.762	.465	.514	.342	13.20
Semantic Entropy	.181	.078	.521	-.085	-.039	.490	.473	.744	.673	.546	.358	11.50
SAR	.107	.087	.491	.217	.069	.496	.472	<u>.781</u>	.690	.545	.396	9.30
LUQ	.104	.114	.261	.268	.140	.411	.430	.755	.503	.451	.344	13.50
Semantic Density	-.003	.073	.323	.210	.241	.512	.520	.712	.475	.405	.347	13.20
Factoscope	.090	.063	.088	.492	-.093	-.056	.480	.289	.542	.084	.198	16.40
SAPLMA	.318	.019	.600	.240	.375	-.005	.535	.601	.535	.604	.382	10.70
MIND	.292	.098	.608	<u>.608</u>	<u>.511</u>	.345	.524	.528	<u>.782</u>	.702	.500	7.10
Sheeps	.304	.080	.638	.561	.397	.358	.439	.551	.733	<u>.756</u>	.482	9.00
LookBackLens	**.475**	<u>.194</u>	**.672**	.543	.481	.465	**.666**	.685	.750	.712	<u>.564</u>	<u>5.00</u>
TAD	<u>.462</u>	**.219**	<u>.643</u>	**.848**	**.575**	**.555**	<u>.641</u>	.773	**.812**	**.769**	**.630**	**1.60**

Table 5: PRR↑ for Gemma 9b v2 model for various tasks for the considered sequence-level methods. Warmer color indicates better results. The best method is in **bold**, the second best is <u>underlined</u>.

UQ Method	SamSum AlignScore	CNN AlignScore	WMT19 Comet	MedQUAD AlignScore	TruthfulQA AlignScore	CoQA AlignScore	SciQ AlignScore	TriviaQA AlignScore	MMLU Acc.	GSM8k Acc.	Mean PRR	Mean Rank
MSP	.394	.148	<u>.582</u>	.421	.210	.490	<u>.661</u>	.706	.508	.455	.457	7.40
Perplexity	.114	.047	.503	.448	.232	.461	.447	.717	.310	.536	.381	12.20
Mean Token Entropy	.036	.049	.487	.460	.251	.435	.302	**.733**	.060	.553	.337	13.20
CCP	.374	.117	.455	.433	.131	.398	.422	.707	.197	.470	.370	13.50
Focus	.156	.056	.503	.494	.196	.356	.500	.643	-.351	.474	.303	15.00
Simple Focus	.317	.093	.570	.350	.250	<u>.513</u>	.639	.718	.449	.490	.439	7.50
Lexical Similarity Rouge-L	.244	.059	.485	.189	.151	.400	.553	.653	.381	.683	.380	12.80
EigenScore	.050	.054	.489	-.003	.089	.426	.643	.643	.364	.709	.346	13.40
EVL NLI Score entail.	.206	.091	.383	.224	.270	.468	.595	.675	.290	.572	.377	12.10
Ecc. NLI Score entail.	.186	.036	.439	.137	.216	.401	.598	.648	.342	.590	.359	14.10
DegMat NLI Score entail.	.214	.091	.418	.234	.263	.419	.546	.699	.319	.593	.380	12.00
Semantic Entropy	.262	.081	.514	.179	.189	.458	.589	.674	.252	.564	.376	12.60
SAR	.238	.076	.515	.342	.224	.475	.634	.707	.333	.708	.425	8.90
LUQ	.123	.075	.314	.093	.278	.423	.543	.682	.321	.607	.346	13.30
Semantic Density	.118	.024	.336	.090	.271	.460	.611	.695	.294	.600	.350	13.50
Factoscope	.064	.016	.134	.476	.038	.205	.447	.467	.821	-.368	.230	17.10
SAPLMA	.283	.030	.416	.437	.316	-.035	.442	.519	.432	.643	.348	13.20
MIND	.316	.124	.308	.527	.369	.489	.640	.639	.890	.783	.508	7.40
Sheeps	.395	**.180**	.515	.547	.387	.380	.429	.704	<u>.900</u>	**.837**	.527	6.50
LookBackLens	**.445**	<u>.159</u>	.571	<u>.597</u>	<u>.398</u>	.434	**.703**	.708	.848	.753	<u>.562</u>	<u>3.50</u>
TAD	<u>.434</u>	.140	**.607**	**.732**	**.468**	**.515**	.648	<u>.728</u>	**.904**	<u>.825</u>	**.600**	**1.80**

Table 6: PRR↑ for Qwen 7b v2.5 model for various tasks for the considered sequence-level methods. Warmer color indicates better results. The best method is in **bold**, the second best is <u>underlined</u>.

A.2 Results Using the ROC-AUC Metric

The results with the ROC-AUC metric are presented in Tables 7 to 9. We obtain discrete versions of the generation quality metrics by thresholding the original continuous values. The thresholds were empirically determined as 0.3 for SamSum and CNN/DailyMail; 0.5 for MedQUAD, TruthfulQA, CoQA, SciQ, and TriviaQA; and 0.85 for WMT19. The results align with the trends observed in the PRR metric. Overall, TAD outperforms the second-best method (LookBackLens) by 1.1% for LLaMa-3.1 8B, 2.4% for Gemma-2 9B, and 0.4% for Qwen-2.5 7B on average across all datasets.

UQ Method	SamSum AlignScore	CNN AlignScore	WMT19 Comet	MedQUAD AlignScore	TruthfulQA AlignScore	CoQA AlignScore	SciQ AlignScore	TriviaQA AlignScore	MMLU Acc.	GSM8k Acc.	Mean ROC-AUC	Mean Rank
MSP	.622	.557	.726	.841	.710	.655	.776	.809	.771	.672	.714	8.50
Perplexity	.518	.491	.722	.856	.658	.665	.678	.804	.741	.652	.679	12.70
Mean Token Entropy	.512	.485	.728	.842	.658	.662	.669	.815	.619	.664	.665	13.70
CCP	.634	.539	.671	.816	.619	.633	.704	.824	.709	.678	.683	10.90
Focus	.577	.514	.708	.840	.656	.624	.785	.793	.642	.668	.681	12.80
Simple Focus	.630	.551	.738	.804	.657	.671	.804	.821	.758	.669	.710	8.00
Lexical Similarity Rouge-L	.559	.534	.679	.566	.536	.684	.803	.783	.642	.683	.647	12.90
EigenScore	.482	.537	.673	.497	.530	.657	.746	.766	.612	.651	.615	17.00
EVL NLI Score entail.	.568	.532	.630	.564	.600	.700	.765	.822	.640	.638	.646	13.50
Ecc. NLI Score entail.	.503	.492	.648	.562	.584	.684	.767	.794	.642	.655	.633	15.30
DegMat NLI Score entail.	.570	.533	.636	.571	.601	.701	.798	.828	.647	.649	.654	11.40
Semantic Entropy	.558	.534	.693	.625	.565	.649	.722	.792	.624	.696	.646	14.30
SAR	.581	.536	.717	.676	.554	.683	.813	.821	.676	.687	.675	10.20
LUQ	.590	.529	.618	.548	.591	.687	.759	.820	.647	.606	.640	14.10
Semantic Density	.543	.520	.638	.679	.642	.720	.785	.829	.622	.614	.659	12.40
Factoscope	.529	.531	.592	.751	.571	.513	.698	.705	.820	.558	.627	16.70
SAPLMA	.652	.516	.792	.872	.593	.509	.741	.728	.733	.713	.685	11.40
MIND	.648	.563	.748	.924	.708	.654	.813	.785	**.884**	.795	.752	6.20
Sheeps	.671	.581	.778	.913	.674	**.746**	**.827**	.827	.881	**.816**	.771	2.70
LookBackLens	**.718**	**.588**	**.820**	.924	.734	.701	.826	.778	.874	.780	.774	3.80
TAD	.710	.575	.811	**.956**	**.764**	.684	.823	**.842**	.879	.805	**.785**	**2.50**

Table 7: ROC-AUC↑ of UQ methods for the Llama-3.1 8b model. Warmer color indicates better results. The best method is in **bold**, the second best is underlined.

UQ Method	SamSum AlignScore	CNN AlignScore	WMT19 Comet	MedQUAD AlignScore	TruthfulQA AlignScore	CoQA AlignScore	SciQ AlignScore	TriviaQA AlignScore	MMLU Acc.	GSM8k Acc.	Mean ROC-AUC	Mean Rank
MSP	.693	.523	.732	.690	.662	.698	.786	.863	.846	.681	.718	8.10
Perplexity	.519	.492	.733	**.926**	.638	.703	.664	.867	.840	.634	.702	10.80
Mean Token Entropy	.503	.483	.734	.926	.627	.689	.658	**.874**	.841	.657	.699	11.50
CCP	.662	.511	.679	.764	.608	.669	.711	.857	.816	.699	.698	12.40
Focus	.554	.493	.721	.913	.664	.676	.774	.842	.830	.664	.713	11.10
Simple Focus	.664	.530	.761	.753	.643	.712	.806	.865	.838	.671	.724	7.60
Lexical Similarity Rouge-L	.530	.547	.699	.544	.494	.692	.724	.851	.758	.700	.654	13.50
EigenScore	.574	.540	.623	.613	.441	.680	.683	.820	.737	.630	.634	16.90
EVL NLI Score entail.	.574	.540	.651	.585	.556	.696	.732	.865	.760	.680	.664	12.60
Ecc. NLI Score entail.	.502	.515	.663	.647	.515	.692	.745	.847	.758	.700	.658	13.80
DegMat NLI Score entail.	.576	.541	.657	.566	.552	.694	.743	.867	.747	.695	.664	12.40
Semantic Entropy	.576	.544	.704	.500	.493	.679	.706	.839	.779	.727	.655	13.50
SAR	.545	.551	.735	.680	.572	.698	.732	.872	.799	.710	.689	9.20
LUQ	.579	.559	.642	.618	.616	.681	.689	.865	.756	.657	.666	13.10
Semantic Density	.499	.544	.661	.655	.648	**.734**	.772	.858	.697	.634	.670	12.30
Factoscope	.552	.527	.529	.865	.456	.493	.715	.640	.718	.523	.602	17.20
SAPLMA	.673	.505	.831	.808	.703	.499	.772	.776	.738	.760	.707	10.60
MIND	.638	.547	.826	.812	.748	.660	.766	.756	.847	.821	.742	7.70
Sheeps	.663	.526	.831	.791	.709	.668	.764	.782	.806	**.853**	.739	9.10
LookBackLens	**.758**	.596	**.845**	.807	.736	.697	**.852**	.816	.828	.817	.775	5.20
TAD	.744	**.604**	.820	.925	**.773**	.714	.833	.866	**.863**	.847	**.799**	**2.40**

Table 8: ROC-AUC↑ for Gemma 9b v2 model for various tasks for the considered sequence-level methods. Warmer color indicates better results. The best method is in **bold**, the second best is underlined.

UQ Method	SamSum AlignScore	CNN AlignScore	WMT19 Comet	MedQUAD AlignScore	TruthfulQA AlignScore	CoQA AlignScore	SciQ AlignScore	TriviaQA AlignScore	MMLU Acc.	GSM8k Acc.	Mean ROC-AUC	Mean Rank
MSP	.638	.544	.733	.843	.580	.678	.797	.819	.813	.670	.712	9.70
Perplexity	.561	.506	.755	.848	.638	.674	.735	.822	.697	.746	.698	11.50
Mean Token Entropy	.540	.512	.760	.847	.652	.668	.711	.832	.590	.756	.687	12.20
CCP	.618	.536	.688	.840	.543	.639	.751	.821	.673	.679	.679	13.00
Focus	.590	.529	.725	.837	.603	.641	.746	.799	.439	.676	.659	14.70
Simple Focus	.627	.537	.751	.777	.618	.698	.798	.826	.772	.693	.709	8.60
Lexical Similarity Rouge-L	.610	.532	.699	.556	.551	.676	.770	.795	.665	.787	.664	12.90
EigenScore	.551	.511	.687	.476	.557	.671	.775	.781	.654	.803	.647	15.10
EVL NLI Score entail.	.592	.527	.646	.628	.638	.701	.771	.822	.643	.713	.668	12.50
Ecc. NLI Score entail.	.574	.515	.680	.551	.623	.678	.766	.799	.655	.725	.657	14.60
DegMat NLI Score entail.	.596	.529	.656	.632	.637	.696	.769	.827	.645	.727	.671	12.40
Semantic Entropy	.608	.548	.686	.713	.592	.670	.758	.798	.623	.734	.673	13.60
SAR	.610	.545	.722	.699	.609	.698	.790	.820	.685	.797	.697	9.10
LUQ	.550	.532	.637	.507	.653	.699	.761	.817	.674	.730	.656	13.10
Semantic Density	.578	.507	.677	.602	.604	**.731**	.798	.828	.621	.739	.668	11.90
Factoscope	.506	.513	.540	.836	.521	.585	.706	.716	.909	.409	.624	17.70
SAPLMA	.659	.527	.667	.844	.654	.501	.720	.749	.709	.761	.679	12.30
MIND	.674	.574	.682	.804	.676	.722	.795	.812	.939	.882	.756	6.20
Sheeps	.670	**.596**	.760	.834	**.702**	.695	.776	**.846**	.945	**.885**	.771	4.00
LookBackLens	**.719**	.589	.772	.883	.682	.706	**.843**	.827	.928	.839	.779	**2.70**
TAD	.707	.577	**.790**	**.915**	.702	.694	.787	.837	**.945**	.879	**.783**	3.20

Table 9: ROC-AUC↑ for Qwen 7b v2.5 model for various tasks for the considered sequence-level methods. Warmer color indicates better results. The best method is in **bold**, the second best is underlined.

A.3 Generalization to Out-of-Domain Tasks

In this experiment, we examine how our approach can be generalized on the unseen datasets. For each target dataset, we construct a general QA training dataset by sampling 300 instances from the training datasets from each of other QA datasets. Thus, we evaluate TAD that is not trained on the target dataset. We conduct experiments on one dataset from each task: SamSum, CNN, WMT19, MedQUAD, CoQA, SciQ, MMLU, and GSM8k. We compare the results with the baseline MSP method.

Table 3 presents the performance of the supervised methods against the MSP baseline on QA tasks, while Table 10 presents the results when trained on QA datasets and evaluated on summarization and translation tasks. The results demonstrate that TAD consistently outperforms baselines on unseen QA domains, while its generalization across diverse task types remains limited.

UQ Method	SamSum AlignScore	CNN AlignScore	WMT19 Comet	Mean PRR
MSP	**.298**	**.157**	**.569**	**.342**
Factoscope	.077	.023	.131	.077
SAPLMA	.045	.021	-.250	-.061
MIND	.077	<u>.048</u>	.174	<u>.099</u>
Sheeps	<u>.104</u>	-.021	.157	.080
LookBackLens	-.026	-.032	.018	-.013
TAD	.035	.003	<u>.234</u>	.091

Table 10: PRR↑ for Llama 8b v3.1 model for summarization and translation tasks for the considered supervised sequence-level methods trained on the general QA dataset. Unsupervised methods are not included as their performance is not dependent of the training data. Warmer colors indicate better results. The best method is in **bold**, and the second best one is <u>underlined</u>.

A.4 Replacing Attention Weights with Layer Integrated Gradients (LIG) Features in TAD

In this part, we expand our experiments by incorporating the use of Layer Integrated Gradients (LIG; Sundararajan et al. (2017)) as an alternative or addition to attention weights in the TAD method. The LIG features were computed using Captum's (Kokhlikyan et al., 2020) `attribute` method, where for each predicted token t_i, attributions were calculated with respect to the input and previously generated tokens. Attribution vectors were aggregated across all layers and aligned to match the shape of the attention matrices.

The motivation behind this experiment was to assess whether attribution-based interpretability features, such as LIG, which estimate token importance with respect to model outputs, could serve as a more semantically grounded alternative to raw attention weights. Given the increasing critique of attention as explanation, it was natural to test whether LIG-based representations improve uncertainty modeling.

Table 11 compares the original TAD method with two modified variants: *TAD (LIG)*, which replaces attention weights entirely with LIG attributions, and *TAD (MIX)*, which concatenates LIG attributions with the original attention weights. The results demonstrate that the *TAD (LIG)* method performs the worst across all tasks, particularly on TruthfulQA and SamSum, where it achieves notably low PRR scores. While *TAD (MIX)* significantly outperforms the LIG-only variant, the original TAD method remains superior, achieving the highest average performance across all datasets.

The experiment demonstrates that LIG attributions, while interpretable and semantically grounded, are ineffective as a replacement for attention weights for uncertainty quantification. Furthermore, combining attention weights with LIG attributions can worsen the performance of the TAD method.

UQ Method	SamSum AlignScore	TruthfulQA AlignScore	CoQA AlignScore	SciQ AlignScore	TriviaQA AlignScore	MMLU Acc.
TAD (LIG)	0.246	.252	0.447	0.553	0.669	0.729
TAD (MIX)	<u>0.392</u>	<u>.521</u>	**0.510**	<u>0.633</u>	<u>0.716</u>	<u>0.789</u>
TAD	**0.431**	**.565**	<u>0.509</u>	**0.644**	**0.737**	**0.806**

Table 11: PRR↑ for Llama 8b v3.1 model for various modifications of the TAD method using the LIG features. The best method is in **bold**, the second best is <u>underlined</u>.

A.5 Ablation Studies

Here, we present ablation studies for various numbers of the preceding tokens, different features, and the impact of various layers for the TAD method.

UQ Method	Aggregation	SamSum AlignScore	CNN AlignScore	WMT19 Comet	MedQUAD AlignScore	TruthfulQA AlignScore	CoQA AlignScore	SciQ AlignScore	TriviaQA AlignScore	MMLU Acc.	GSM8k Acc.	Mean PRR	Mean Rank
TAD (LinReg)	$\frac{1}{K}\sum_{k=1}^{K} p_k$	**.431**	.215	**.612**	**.662**	**.565**	**.543**	.542	.757	.806	**.682**	**.581**	**1.80**
TAD (LinReg)	$\sum_{k=1}^{K} \log p_k$	.348	**.245**	.462	.307	.450	.509	**.644**	.737	.816	.605	.512	2.80
TAD (MLP)	$\frac{1}{K}\sum_{k=1}^{K} p_k$	.402	.208	.602	.591	.482	.526	.491	**.764**	.814	.645	.552	2.40
TAD (MLP)	$\sum_{k=1}^{K} \log p_k$	.375	.239	.397	.222	.461	.482	.626	.746	**.818**	.522	.489	3.00

Table 12: Comparison of various considered regression models and aggregation strategies for TAD (PRR↑, Llama 8b v3.1 model). Warmer colors indicate better results.The best method is in **bold**, the second best is underlined.

UQ Method	SamSum AlignScore	CNN AlignScore	WMT19 Comet	MedQUAD AlignScore	TruthfulQA AlignScore	GSM8k Acc.	Mean PRR
TAD (1 tokens)	.425	**.228**	.602	.570	.519	.659	.501
TAD (2 tokens)	.424	.224	.606	.596	.537	.679	.511
TAD (5 tokens)	.397	.219	**.618**	.628	.556	**.687**	.517
TAD (10 tokens)	**.431**	.215	.612	**.662**	**.565**	.682	**.528**

Table 13: PRR↑ for Llama 8b v3.1 model for various tasks for the various number of preceding tokens for the TAD method. Warmer color indicates better results. The best method is in **bold**, the second best is underlined.

UQ Method	SamSum AlignScore	CNN AlignScore	WMT19 Comet	MedQUAD AlignScore	TruthfulQA AlignScore	GSM8k Acc.	Mean PRR
TAD (Sequence-level)	.455	**.252**	**.650**	.618	.520	.608	.517
TAD	**.465**	.211	.622	**.696**	**.565**	**.682**	**.540**

Table 14: PRR↑ for the modifications of the TAD method for the Llama-3.1 8b model. The best method is in **bold**, the second best is underlined.

UQ Method	SamSum AlignScore	CNN AlignScore	WMT19 Comet	MedQUAD AlignScore	TruthfulQA AlignScore	CoQA AlignScore	SciQ AlignScore	TriviaQA AlignScore	MMLU Acc.	GSM8k Acc.	Mean PRR
TAD (probs.)	.178	.086	.411	.437	.270	.444	.567	.683	.668	.374	.412
TAD (attention)	.426	.212	.611	**.670**	**.566**	.480	.632	.712	.804	.673	.579
TAD (attention+probs.)	**.431**	**.215**	**.612**	.662	.565	**.509**	**.644**	**.737**	**.806**	**.682**	**.586**

Table 15: PRR↑ for Llama 8b v3.1 model for various tasks for different features for the TAD method. Warmer color indicates better results. The best method is in **bold**, the second best is underlined.

UQ Method	SamSum AlignScore	CNN AlignScore	WMT19 Comet	MedQUAD AlignScore	TruthfulQA AlignScore	CoQA AlignScore	SciQ AlignScore	TriviaQA AlignScore	MMLU Acc.	GSM8k Acc.	Mean PRR
TAD (1 step)	.107	.043	.281	.057	.168	.421	.499	.677	.397	.285	.294
TAD (2 step)	**.431**	**.215**	**.612**	**.662**	**.565**	**.509**	**.644**	**.737**	**.806**	**.682**	**.586**

Table 16: PRR↑ for Llama 8b v3.1 model for various tasks for the different number of learning steps for the TAD method. Warmer color indicates better results. The best method is in **bold**.

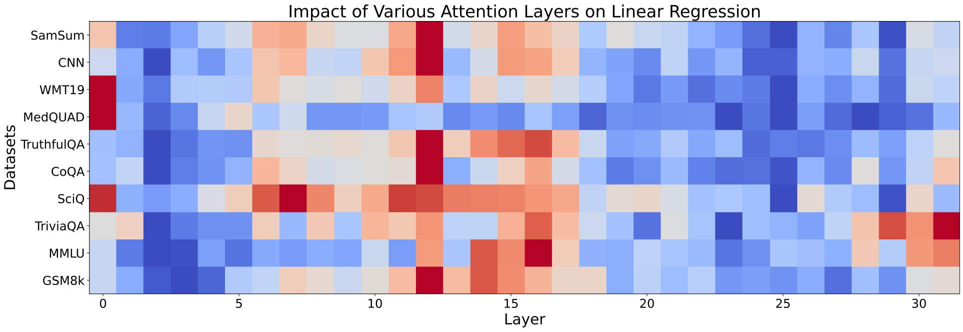

Figure 3: Normalized average weights of linear regression for different attention layers in the TAD method across the considered datasets. Warmer color indicates a higher impact on the TAD performance.

B Computational Resources and Efficiency

All experiments were conducted on a single NVIDIA H100 GPU. On average, training a single model across all datasets took over 750 GPU hours, while inference on the test set took 260 GPU hours.

C Hyperparameters

C.1 Optimal Hyperparameters for TAD

The optimal hyperparameters for TAD for various considered regression models and different aggregation strategies are presented in Tables 17 to 19 for Llama-3.1 8b, Gemma-2 9b, and Qwen-2.5 7b models respectively. These hyperparameters are obtained using cross-validation with five folds using the training dataset. We train a regression model on $k-1$ folds of the training dataset and estimate uncertainty on the remaining fold. The optimal hyperparameters are selected according to the best average PRR for AlignScore. Finally, we use these hyperparameters to train the regression model on the entire training set.

The hyperparameter grid for the linear regression is the following:
L2 regularization: [1e+1, 1, 1e-1, 1e-2, 1e-3, 1e-4].

The hyperparameter grid for the MLP is the following:
Num. of layers: [2, 4];
Num. of epochs: [10, 20, 30];
Learning rate: [1e-5, 3e-5, 5e-5];
Batch size: [64, 128].

UQ Method	Aggregation	SamSum	CNN	WMT19	MedQUAD	TruthfulQA	CoQA	SciQ	TriviaQA	MMLU	GSM8k
TAD (MLP)	$\frac{1}{K}\sum_{k=1}^{K} p_k$	4, 30, 1e-05, 0, 128	4, 30, 3e-05, 0, 128	4, 30, 3e-05, 0, 128	4, 30, 1e-05, 0, 128	4, 30, 5e-05, 0, 128	4, 30, 3e-05, 0, 64	2, 30, 5e-05, 0, 128	4, 30, 3e-05, 0, 128	4, 30, 5e-05, 0, 128	4, 30, 1e-05, 0, 128
TAD (MLP)	$\sum_{k=1}^{K} \log p_k$	4, 30, 5e-05, 0, 64	4, 30, 1e-05, 0, 128	2, 20, 5e-05, 0, 64	4, 30, 5e-05, 0, 64	4, 30, 5e-05, 0, 64	2, 30, 5e-05, 0, 64	4, 30, 5e-05, 0, 128	4, 30, 3e-05, 0, 128	4, 30, 5e-05, 0, 64	4, 30, 3e-05, 0, 128
TAD (LinReg)	$\frac{1}{K}\sum_{k=1}^{K} p_k$	1	10.0	1	0.01	1	1	0.001	10.0	1	10.0
TAD (LinReg)	$\sum_{k=1}^{K} \log p_k$	1	1	0.0001	0.001	0.1	10.0	10.0	1	1	0.01

Table 17: Optimal values of the hyper-parameters for the TAD methods for the Llama 8b v3.1 model.

UQ Method	SamSum	CNN	WMT19	MedQUAD	TruthfulQA	CoQA	SciQ	TriviaQA	MMLU	GSM8k
TAD (LinReg)	10.0	10.0	0.0001	0.0001	1	10.0	1	10.0	10.0	1

Table 18: Optimal values of the hyper-parameters for the final configuration of the TAD method for the Gemma 9b v2 model.

UQ Method	SamSum	CNN	WMT19	MedQUAD	TruthfulQA	CoQA	SciQ	TriviaQA	MMLU	GSM8k
TAD (LinReg)	0.1	0.01	1	0.0001	0.01	10.0	10.0	10.0	1	1

Table 19: Optimal values of the hyper-parameters for the final configuration of the TAD method for the Qwen 7b v2.5 model.

C.2 LLM Generation Hyperparameters

Dataset	Task	Max Input Length	Generation Length	Temperature	Top-p	Do Sample	Beams	Repetition Penalty
SamSum	TS		128					
CNN			128					
WMT19	MT		107					
MedQUAD	QA Long answer		128					
TruthfulQA		-	128	1.0	1.0	False	1	1
GSM8k			256					
CoQA	QA Short answer		20					
SciQ			20					
TriviQA			20					
MMLU	MCQA		3					

Table 20: Values of the text generation hyper-parameters for all LLMs used in our experiments.

D Dataset Statistics

Statistics about the datasets are provided in Table 21. For TS, we experiment with CNN/DailyMail (See et al., 2017) and SamSum (Gliwa et al., 2019). For the long answer QA task, we use MedQUAD (Abacha and Demner-Fushman, 2019), which consists of real medical questions, TruthfulQA (Lin et al., 2022), which consists of questions that some people would answer incorrectly due to a false belief or a misconception, and GSM8k (Cobbe et al., 2021) with a grade school math questions. For the QA task with short answers, we follow previous work on UQ (Kuhn et al., 2023; Duan et al., 2024; Lin et al., 2024) and we use three datasets: SciQ (Welbl et al., 2017), CoQA (Reddy et al., 2019), and TriviaQA (Joshi et al., 2017). For multiple-choice QA, we use MMLU (Hendrycks et al., 2021), a widely used benchmark for evaluating LLMs. For MT, we use WMT19 (Barrault et al., 2019), focusing on translations from German to English.

Task	Dataset	N-shot	Train texts for TAD	Evaluation texts
Text Summarization	CNN/DailyMail	0	2,000	2,000
	SamSum	0	2,000	819
MT	WMT19 De-En	0	2,000	2,000
QA Long answer	MedQUAD	5	700	2,000
	TruthfulQA	5	408	409
	GSM8k	5	700	1,319
QA Short answer	SciQ	0	2,000	1,000
	CoQA	all preceding questions	2,000	2,000
	TriviaQA	5	2,000	2,000
MCQA	MMLU	5	2,000	2,000

Table 21: Statistics about the datasets used for evaluation.

E Generating Training Data for TAD

Algorithm 1: Generating training data for TAD

Data: Input prompt $\mathbf{x}_k$, LLM generation $\mathbf{y}_k = t_{1:n_k}$, token probabilities $p(t_i \mid t_{<i}, \mathbf{x}_k)$, number of preceding tokens N, vector of LLM attention weights $a_{i,i-l}$ from the $(i-l)$-th token to the i-th token from all layers and heads, and step of the training procedure j

Result: Feature vectors z_i^k, $k = 1 \dots K$, $i = 2 \dots n_k$

`// Estimate unconditional probability for the first token`

$\hat{p}_k(t_1) = \mathrm{sim}(\mathbf{y}_k, \mathbf{y}_k^*)$;

for $i \leftarrow 2$ **to** n_k **do**

    `// Construct token-level features`

    $z_i^k \leftarrow \bigoplus_{l=1}^{\min\{N,i-1\}} \Big[p(t_{i-l} \mid \mathbf{t}_{<i-l}, \mathbf{x}_k),\ \hat{p}_k(t_{i-l}),\ a_{i,i-l}\Big] \oplus \Big[p(t_i \mid \mathbf{t}_{<i}, \mathbf{x}_k)\Big]$;

    `// If` $N > i-1$`, we pad` z_i^k `with zeros to ensure they have the same length`

    if $i - 1 < N$ **then**

        $z_i^k \leftarrow z_i^k \oplus \mathbf{0}_{(2+|a_{i,i-l}|)(N-i-1)}$;

    `// Estimate token-level unconditional probability`

    if $j == 1$ **then**

        `// On the first training step, we use ground truth`

        $\hat{p}_k(t_i) = \mathrm{sim}(\mathbf{y}_k, \mathbf{y}_k^*)$;

    else

        `// On the next training steps, we use trained function` $C(\cdot)$

        $\hat{p}_k(t_i) = C(z_i^k)$;

return z_i^k, $k = 1 \dots K$, $i = 2 \dots n_k$;