Factual Confidence of LLMs: on Reliability and Robustness of Current Estimators

Anonymous ACL submission

Abstract

Large Language Models (LLMs) tend to be unreliable on fact-based answers. To address this problem, NLP researchers have proposed a range of techniques to estimate LLM's confidence over facts. However, due to the lack of a systematic comparison, it is not clear how the different methods compare to one other. To fill this gap, we present a rigorous survey and empirical comparison of estimators of factual confidence. We define an experimental framework allowing for fair comparison, covering both fact-verification and QA. Our experiments across a series of LLMs indicate that trained hidden-state probes provide the most reliable confidence estimates; albeit at the expense of requiring access to weights and supervision data. We also conduct a deeper assessment of the methods, in which we measure the consistency of model behavior under meaning-preserving variations in the input. We find that the factual confidence of LLMs is often unstable across semantically equivalent inputs, suggesting there is much room for improvement for the stability of models' parametric knowledge.

1 Introduction

A major problem of Large Language Models (LLMs) is that they do not always generate truthful information. Models can hallucinate by convincingly reporting information that is actually false or they are not confident about, or provide factual answers only when prompted in a certain way (Elazar et al., 2021; Wang et al., 2023a; Lin et al., 2022b; Ji et al., 2023; Luo et al., 2023). This behavior can be severely harmful, especially given the current explosion of LLM usage: a lack of truthfulness can lead to spread of misinformation and breaches to the user trust (Weidinger et al., 2021; Bender et al., 2021; Evans et al., 2021; Tamkin et al., 2021). Having a reliable estimate of the model confidence over a fact—the degree to which it is expected to have accurate factual knowledge with respect to an input—is key for mitigating this problem (Geng et al., 2023; Tonmoy et al., 2024).

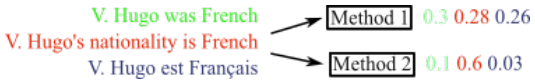

Figure 1: First, we estimate factual confidence using a range of methods. Then, we test whether semantics-preserving input variants yield consistent estimates (Method 1) or not (Method 2).

Recently, a number of papers proposed methods to estimate an LLM' factual confidence (Burns et al. 2022; Lin et al. 2022a; Kuhn et al. 2023; Azaria and Mitchell 2023; Pacchiardi et al. 2023, among others). However, none of them establishes a unified experimental framework to compare methods. This leaves open questions regarding how aligned the methods are in their estimates, and which are the most reliable to apply across LLMs.

We aim to fill this gap by presenting a survey on LLM factual confidence estimation, and performing a systematic empirical comparison of the methods proposed. We first categorize existing methods into groups of related approaches (e.g., trained probes, verbalized confidence). We then introduce an experimental framework enabling a comparison across methods under fixed experimental conditions (Figure 1). Our work is guided by explicit definitions of two ways of measuring factual confidence: 1) the probability of a statement to be true, noted $P(\text{True})$, and 2) the probability of yielding a truthful answer to a query, noted $P(\text{I know})$ (Kadavath et al., 2022). These align to a fact-verification (Thorne et al., 2018; Azaria and Mitchell, 2023) and Question Answering (QA) (Kadavath et al., 2022; Yin et al., 2023) setups, both adopted as test methods.

We study the reliability of the confidence estimation methods across eight publicly available LLMs. Our results indicate that prompting-based methods are less reliable than supervised-probing, although the latter requires training data and access to model

weights. For instruction-tuned LLMs, some non-trained methods provide viable alternatives.

We argue that all methods for estimating factual confidence can ultimately lead to misleading conclusions if only tested on a single way of asserting a fact: An LLM may seem to know a fact given an input, but then contradict itself given an alternative writing of the same fact (Elazar et al., 2021; Kassner et al., 2021; Lin et al., 2022b; Qi et al., 2023; Kuhn et al., 2023). In our experiments, we find evidence of such instability, suggesting that the way LLMs encode facts does not always represent abstractions over diverse input variations.

In summary, this paper provides the following contributions: 1) A survey of the literature on LLM factual confidence estimation; 2) An experimental framework enabling a fair comparison across proposed methods;[1] 3) Insights about the reliability and robustness of such methods, providing recommendations for NLP practictioners; 4) Insights about the consistency of factual confidence across semantically equivalent inputs.

2 Factual Confidence: Key Concepts

2.1 Fact

We take a *fact* to be a piece of information that accurately represents a world state.[2] A natural-language statement is *truthful*—or *factual*—if its meaning reports a state of affairs that is supported by a true fact: e.g., "Paris is a city in France." is truthful as the city of Paris is indeed located in France. Facts and natural-language statements are not linked by a one-to-one relation: The same fact can be declared with multiple statements, varying on the surface level, but sharing the same meaning.

For this reason, one's confidence in a fact should be consistent across meaning-preserving linguistic variations, such as paraphrases or translations of a statement: If we are certain that "Paris is a city in France" is true, we will not doubt that its paraphrase "Paris is a French city" or its translation in French (if we understand French) are also true.

2.2 Factual Confidence

We distinguish between two facets of factual confidence of LLMs, following Kadavath et al. (2022):

P(**True**), shortened as $P(\mathrm{T})$: the degree to which a model considers likely that a fact stated in the input is true; e.g., "Paris is the capital of France" should get a high $P(\mathrm{T})$ as it is truthful, while "Sidney is the capital of France" should get a low $P(\mathrm{T})$. To estimate $P(\mathrm{T})$ scores we need to pass a statement in the input, which is evaluated in its truthfulness: this is in line with the setup of **fact verification** (Thorne et al., 2018).

P(**I Know**) shortened as $P(\mathrm{IK})$: the degree to which a model considers likely that it will return the correct answer to an input querying about a fact. For instance, we can compute $P(\mathrm{IK})$ in a **QA** setup passing a question as input—e.g., "What is the capital of France?". If confident to know the true answer, $P(\mathrm{IK})$) should be high; it should instead be low in case of uncertainty. In contrast to $P(\mathrm{T})$, $P(\mathrm{I\ Know})$ is estimated without stating the fact in the input, but rather expecting a factual answer by the model complementing the the query.

$P(\mathrm{T})$ and $P(\mathrm{IK})$ are both telling of the underlying factual confidence of an LLM. However, depending on the data format—e.g., statements vs. questions—or task of interest—e.g., fact verification vs. QA—focusing on one of the measures is more suitable. Previous works introducing methods to estimate factual confidence have typically addressed only one of the two measures. However, as we demonstrate with our experimental framework, most method can be adapted to estimate both $P(\mathrm{T})$ and $P(\mathrm{IK})$, although in practice they may not be equally reliable in each setup.

2.3 Robustness of Factual knowledge

We work from the hypothesis previously voiced by Petroni et al. (2019) that a language model's factual knowledge may stem from encoding facts in its weights—*parametric memory*—as an abstraction over the linguistic input in the training data.

However, such human-like robustness and abstraction ability cannot be taken for granted (Mitchell and Krakauer, 2023; Mahowald et al., 2023; Bender and Koller, 2020). Testing for consistency to meaning-preserving variations of an input is key to distinguish whether a model has encoded a fact as an abstraction over linguistic forms, as opposed to memorizing statements asserting the fact (Carlini et al., 2022). For instance, if a model has a robust encoding in its parametric memory of what the capital of France is, it should provide the same answer to "What is

[1] We plan to release our code and data upon publication.

[2] For simplicity, in this work, we restrict our focus to minimal, atomic facts, in the sense that they do not involve a combination of subfacts; e.g., "The Louvre is in Paris" as opposed to "The Louvre is in Paris, which is in France".

	Black-box	Trained	Prompt-based	Scores for
Trained Probe	No	Yes	No	$P(\text{T})$ & $P(\text{IK})$
Verbalisation	Yes	No	Yes	$P(\text{T})$ & $P(\text{IK})$
Surrogate Token Probability	Yes (*)	No	Yes	$P(\text{T})$ & $P(\text{IK})$
Average Sequence Probability	Yes (*)	No	No	$P(\text{T})$ & $P(\text{IK})$
Consistency	Yes	No	No	$P(\text{IK})$

Table 1: Differences across the methods for measuring factual confidence. *Black-box* marks methods which do not rely on access to model's weights; (*) denotes the possibility to use sampling if token probabilities are not available.

the capital of France?", "What is the name of the French capital city?" or any other rewording. Prior works already provided evidence that models may not always act consistently across semantically equivalent inputs (Elazar et al., 2021; Kassner et al., 2021; Ohmer et al., 2023; Qi et al., 2023). However, this has not been investigated yet in relation to the degree of factual confidence.

3 Factual Confidence: Survey of Methods

Based on a review of the research area, we identify 5 groups of existing methods to estimate factual confidence, which we discuss in the following subsections. In Table 1, we provide an overview of the functional differences among these methods.

3.1 Sequence Probability

This methodology uses the averaged probabilities, assigned to a sequence of output tokens, to estimate factual confidence. It has been applied as a general estimator of a model's confidence over an output in various domains (Gal and Ghahramani, 2016; Guo et al., 2017; Fomicheva et al., 2020; Xiong et al., 2023a). In the context of factual knowledge, sequence probability has been applied both in cloze tasks and QA setups (Jiang et al., 2020; Yin et al., 2023), which corresponds to measuring $P(\text{IK})$.

Gal and Ghahramani (2016) showed that sequence probabilities produce unreliable, specifically over-confident, estimates; it is thus used mainly as a weak baseline. This is not surprising as by focusing on the sequence probability, we target confidence over *how* a claim is made, rather that confidence about the claim itself Lin et al. (2022a).

3.2 Verbalized Confidence

In the *verbalized confidence* method (Xiong et al., 2023b), the model is directly prompted to report its confidence level (e.g., "How confident are you that the answer is correct?"). This method has been proposed as a general way to probe for the confidence of a LLM over its answers. Lin et al. (2022a) find that this method provides well-calibrated and surprisingly accurate estimates for highly capable models like GPT4 (OpenAI, 2023). Additionally, Tian et al. (2023) show that finetuning a model for human preference (RLHF) (Ouyang et al., 2022; Bai et al., 2022) does not reduce calibration, as oppose to the findings in Kadavath et al. (2022). On factual knowledge, Yin et al. (2023) and Tian et al. (2023) applied this method to QA setups following the $P(\text{IK})$ definition of factual confidence.

3.3 Surrogate Token Probability

These methods, extensively studied by Kadavath et al. (2022); Xiong et al. (2023b), can be considered a hybrid approach between the methods presented above. The input prompt asks the model to provide as output specific tokens to report the factuality of the claim in the input; the probabilities assigned to them is used to determine the confidence level. This method can be adapted to measure both P(T) and P(IK) (Kadavath et al., 2022).

3.4 Output Consistency

The *output consistency* method (Wang et al., 2023b)—also known as *self-consistency*—builds on the assumption that a high LLM confidence leads to generating consistent outputs. Given a question or incomplete statement, we sample multiple completions and take the inter-responses consistency as confidence measure: If the same answer is always generated, confidence is high; it is instead lower if the model outputs different responses. A limitation of this method is that, due to its completion setup, it can be used to estimate $P(\text{IK})$, but not $P(\text{T})$.

Manakul et al. (2023) demonstrated the efficacy of this method when applied to factual knowledge, focusing on on GPT models and using output consistency to "fact-check" model responses. Kuhn

et al. (2023) adopted this method, but on a different family of models (OPT) and insisting on the need to cluster outputs that are semantically equivalent as instances of the same answer.

3.5 Layer Output Transformation

The methods listed so far all focused—in one way or another—on model outputs (token scores or generated tokens). By contrast, other approaches focus on internal representations in earlier layers, in the compression stages of the LLM (Voita et al., 2019). Azaria and Mitchell (2023) proposed to train probes to extract factual confidence scores from hidden states, under the argument that such estimates are less subject to surface-level features—how a claim is phrased—and thus more reliable. Their setup is in line with an estimate of $P(\text{T})$. Kadavath et al. 2022 also adopted this method, though focus on a QA setup—estimating $P(\text{IK})$—and training a value head on top of the final layer.

4 Methodology

4.1 Data

We use two publicly available datasets enabling to test factual confidence in both the fact-verification and QA setup. These datasets act as a common baseline to compare the methods, which up to now have not been benchmarked on the same data. For instance, Azaria and Mitchell (2023) test the *Trained probe* on a custom True/False dataset, while Kadavath et al. (2022) use QA datasets.

4.1.1 $P(\text{T})$ in Fact Verification: Lama T-REx

Lama T-REx (Petroni et al., 2019) is a relational dataset made of triplets extracted from Wikipedia <subject, relation, object>, (e.g., <*Victor Hugo, was born in, France*>). We use this dataset to create both true and false statements for estimating $P(\text{T})$. We create false versions of each factual statement, by randomly substituting the object in the triplet with one from the same relation ("Victor Hugo was born in China"). This ensures the right entity type and avoids grammatical errors.

There are 34K triplets in the T-REx dataset. We keep 80% (27K) and as many corresponding false facts for training (only used for *Trained Probe*). This leaves us with 6.8K T-REx true statements and an equal number of false ones for analysis.

4.1.2 $P(\text{IK})$ in QA: PopQA

The PopQA dataset (Mallen et al., 2022) consists of short questions and object-only answers (e.g. "What is George Rankin's occupation? Politician."). The answers are sets of synonymous phrases, lowering the risk of underestimating model's correctness in a QA setup. We chose this dataset since it covers a broad range of entities, with varying degrees of popularity (estimated based on the number of Wikipedia page views).

We use PopQA to test models' factual confidence given a fact-related query, i.e., $P(\text{IK})$. The dataset contains 14K questions: we keep 80% (11K) for training, and 20% (2.8K) for testing. By definition (Section 2.2), the gold labels for $P(\text{IK})$ should indicate if the model outputs a correct answer. Ultimately, the model answer will depend on the decoding strategy; in this work, for simplicity and clarity of interpretation, we use greedy decoding. If the answer is correct, we set the gold $P(\text{IK})$ to 1, else to 0 (more on this in 4.2). As the labels depend on model correctness, the data will have varying proportions of positive labels across models, ranging from $\sim$11% to $\sim$27%.[3]

4.2 Scoring Methods Implementation

We report below the main specifics of our implementation of the methods (details in Appendix A).

4.2.1 Estimating P(T)

Given a statement, we compute $P(\text{T})$ as follows: 1. **Sequence probability:** Average log-probability of the statement's tokens. 2. **Verbalized confidence**: Prompting for the confidence level that the statement is true (Appendix A). 3. **Surrogate token probability**: Log-probability of the "Yes" token following a query on whether the statement is true. 4. **Trained probe**: Following the approach of Azaria and Mitchell (2023), we train a 3-layer fully connected architecture for 10 epochs, passing as input hidden states at layer 24 (better results were found using one of the last layers, but not the very last one). This is a very light network, that can be trained on a CPU in less than 10 minutes. An LLM-specific probe is trained to classify whether a statement is true or false based on the model's hidden representations. We then take the output logit score as an estimate of $P(\text{T})$.

[3]The proportion of $P(\text{IK})$ labels set to True across models is as follows: falcon-40b-instruct: .23, falcon-7b-instruct: .11, falcon-40b: .20, falcon-7b: .15, Mistral-7B: .14, Mixtral-8x7B-Instruct: .27, Mistral-7B-Instruct-v0.2: .16. The questions from PopQA are generally considered hard (ChatGPT: 30% accuracy, SelfRAG (Asai et al., 2023): 55% acc.)

4.3 Estimating P(IK)

To compute the $P(\text{IK})$ estimates, based exclusively on the question, we follow the steps below: 1. **Sequence probability:** Average log-probability of the question's tokens.[4] 2. **Verbalised confidence:** Prompting (see Appendix A) for the confidence level of knowing the answer to the question. 3. **Surrogate token probability:** Log-probability of "Yes" token following a query on knowing the answer to the question. 4. **Trained probe:** We use the same approach as for $P(\text{T})$, but train the probes to predict whether the model's greedy-generated answers will be truthful or not.[5] **Consistency:** First, we prompt the model with the question and sample 10 responses at temperature 1. Then, we compute a matrix of pairwise NLI scores (Laurer et al., 2023) on all generations, and return an average.

4.4 Evaluating Scoring Methods

To evaluate the methods, we use AUPRC—the area under the precision-recall curve, as also done by other works (e.g., Kadavath et al. 2022). Using a metric that considers various decision thresholds enable a robust comparison across methods. The higher AUPRC, the better ranking capability of the method, with cleaner separation between true/false statements or known/unknown facts.

4.5 Models

We study publicly available LLMs, with open access to model weights. This enables us to compare the *Trained Probe* method across all models. We consider a range of models with different sizes (7B to 46.7B), architecture, and training paradigms (instruction-finetuned or not) from the Falcon (Almazrouei et al., 2023), and Mistral (Jiang et al., 2023, 2024) model families (see table 2 for the full list of LLMs and their properties).

4.6 Paraphrasing and Translation

To test methods robustness and to disentangle confidence over a fact from confidence based on a specific wording, we generate semantically equivalent variants of statements/questions from Lama T-REx and PopQA. For each input, we generate 10 paraphrases by prompting Mixtral-8x7B-Instruct-v0.1 (prompt and examples in Appendix B). We remove repetitions and only keep paraphrases that are semantically equivalent to the original input (testing entailment in both directions through an NLI model Laurer et al. 2023). This results in an average of 8 paraphrase per original input.

We also consider translation as another meaning-preserving transformation. Out of the 8 LLMs we test, only the 7b MistralAI models are monolingual for English (this does not necessarily exclude some degree of exposure to other languages). All other models are described as having been trained on French. Furthermore, all models should have lower capabilities in Polish (Falcon models only report a "limited capability", MistralAI models do not mention it at all). We use the AWS translation API[6], manually verifying the quality of a sample of 100 translations.

[4]This implementation captures how surprised the model is by the question, which is linked with expected correctness.

[5]This is a simpler, less computationally expensive version of the approach of Kadavath et al. (2022), where multiple answers are sampled and the probe initially predicts a continuous score—proportion of correct answers in the sampled set.

Names	Size	Open	Arch.	Instruct
Falcon-40B Inst.	40B	✔	Dense	✔
Falcon-40B	40B	✔	Dense	
Falcon-7B Inst.	7B	✔	Dense	✔
Falcon-7B	7B	✔	Dense	
Mixtral Inst.	8x7B	✔	SMoE	✔
Mixtral	8x7B	✔	SMoE	
Mistral Inst.	7B	✔	Dense	✔
Mistral	7B	✔	Dense	

Table 2: The models used in our experiments. *Dense* represents the usual transformer decoder architecture, while SMoE stands for Sparse Mixture of Experts (Shazeer et al., 2017). *Instruct.* models have been instruction fine-tuned. Open models have publicly available weights.

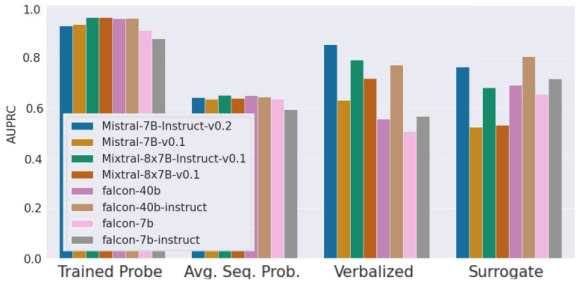

Figure 2: AUPRC scores on T-REx with both true and false statements; P(T).

5 Empirical Comparison of the Methods

5.1 $P(\text{T})$ on Lama T-REx

With each of the 4 methods, we derive estimates of factual confidence for all statements in the Lama T-REx test set, repeating the experiment for each

[6]https://aws.amazon.com/fr/translate/

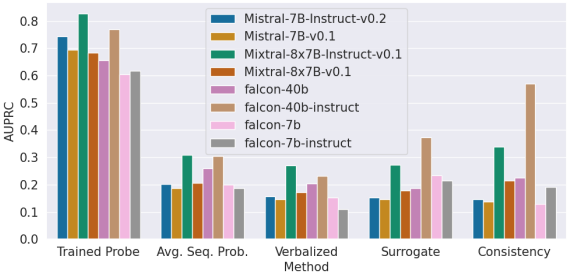

Figure 3: AUPRC scores on PopQA dataset; P(IK).

LLM. We evaluate the reliability of a method by checking whether it yields $P(\mathrm{T})$ scores that can effectively separate the true from the false statements, measured as AUPRC.

We report the results of this analysis in Figure 2. The *Trained Probe* method performs best, outperforming the sequence probabilities by an average AUPRC of .3. Of all methods and models, only the *Verbalised method* is truly competitive, and only for Mistral 7B instruct. Otherwise all methods perform at least .1 AUPRC below the *Trained probe*. The fact that a trained probe applied to the hidden states extracts the most reliable estimates suggests that information about the expected truth value of a statement is better captured in the depth of the network, as opposed to the output scores.

While for *Trained Probe* and *Average Sequence Probability* we note relatively small differences in AUPRC across models, for the *Verbalized* and *Surrogate* methods we see large variation. Concretely, instruction-tuned models always perform better than their counter-parts. This is expected as both methods require to follow instructions in the prompt. Model size also seems to have an effect: all 40B+ models perform better than their 7B counter-parts, with the exception of Mistral-7B-Instruct (this case could be explained by a more effective instruction tuning). Finally, the *Average Sequence probability* method performs consistently above chance (50%), but overall poorly in comparison to other methods, only outperforming other non-trained methods—*Verbalized, Surrogate*—on non-instruction-tuned models.

5.2 $P(\mathrm{IK})$ on PopQA

$P(\mathrm{IK})$ estimates the degree of a model's confidence that its predicted answer will be correct. A good estimator of $P(\mathrm{IK})$ would thus assign high scores to queries which the model answers correctly, and low scores to others. Following this reasoning, for $P(\mathrm{IK})$ we compute the AUPRC scores using binary labels that encode whether the model answer (in our case, generated with greedy decoding) is correct. Note that this way of computing AUPRC—based on a model's *future correctness*—yields an estimate of the method's expected effectiveness when used for hallucination mitigation; that is, to automatically detect when the model should abstain from answering. In this scenario, a method is effective only if its estimates are actually predictive of the correctness of model answers.

The results are reported in Table 3. In this experiment we also study the *Consistency* method, which we omitted from $P(\mathrm{T})$ results because, by design, it cannot be applied to an entire statement. Overall, $P(\mathrm{IK})$ is harder to estimate than $P(\mathrm{T})$, with lower AUPRC results: e.g., The best trained probe is 0.1 below in AUPRC for $P(\mathrm{IK})$ than it is for $P(\mathrm{T})$. This may be due to the complexity of the setup—in QA the confidence is estimated only based on a query, in contrast to fact verification. But it is may also be that the binary *future correctness* labels used for our AUPRC computation introduce some noise: e.g., the model may be genuinely uncertain and still output the correct answer by chance.

The *Trained probe* method is again, by large, the most reliable across all models. With the exception of Falcon-40B instruct, the other methods perform close to or below chance (depending on the model's label distribution, chance level varies between .11 and .27). This indicates that non-trained estimators are generally not reliable for $P(\mathrm{IK})$ despite being frequently used in the literature. Within each method, we observe differences across models—up to a 40% margin. This can be linked to 1) whether a model is instruction-tuned (as noted for P(T)) and 2) the model family—with more reliable scores for MistralAI models than for Falcon models.

5.3 Generalization of the Trained Probe

The results above highlight the Trained probe as the most reliable estimator for factual confidence—both for $P(\mathrm{T})$ and P(IK). However, in those experiments we trained and evaluated the models within the same domain, which leaves open questions about the probe's generalization capabilities. We address this gap by evaluating the model from 5.1, trained to estimate $P(\mathrm{T})$ from Lama T-REx data, on the PopQA dataset converted to test for P(T). Specifically, we re-work the PopQA data for the fact-verification setup by turning question-answer pairs into (evenly distributed) true and false state-

Name	Size	AUPRC	Δ
Falcon	40B	.80	-.16
Falcon Ins.	40B	.81	-.15
Falcon	7B	.66	-.25
Falcon Ins	7B	.59	-.28
Mistral	7B	.62	-.31
Mistral Ins	7B	.75	-.18
Mixtral	46.7B	.78	-.18

Table 3: AUPRC on PopQA test set re-worked as true/false statements, using $P(\mathrm{T})$ estimates from probes trained on Lama T-REx. Δ: difference of AUPRC with respect to that for Lama T-REx data (in-domain).

ments, using the template: *"The answer to [QUESTION] is [ANSWER]"*.[7] We derive estimates for $P(\mathrm{T})$ on such statements using the probes trained on Lama T-REx, and compute AUPRC (Table 3).

Going from in-domain to out-of-domain test data (Lama vs. PopQA), we observe AUPRC differences of min -.15 and max -.31; however, the scores remain in a high range of [.62, .81] indicating substantial generalization. The LLMs for which the probe retain the least and the most reliability are Mistral-7B and Falcon-40B-instruct, respectively. Interestingly, these are also the models getting the least and the most answers right on PopQA in the QA setup (see footnote 3). This suggests that the transferability of the probe may be affected by how challenging the out-of-domain dataset is to the model. In the next sections we provide further evidence of probe generalization by looking at whether and to what extent the AUPRC is affected by input paraphrasing and translating.

6 Robustness to Linguistic Variations

In this section, we apply meaning-preserving linguistic variations to each input statement/qustion to: *1*) assess the robustness of methods, expecting equally reliable estimates across different input formulations, and *2*) investigate the stability of an LLM's encoding of facts, under the view that, if a fact is well abstracted, the factual confidence should be invariant to semantics-preserving changes in the input. We consider two types of input variation: paraphrases and translations.

[7]For true statements we use the gold answers from PopQA dataset. For false statements, we sample alternative answers from the same question class in the dataset; e.g., *The answer to "In which country is Washington?" is "United States of America"* vs. *"South Korea"*.").

6.1 Robustness of Methods

We study method robustness in both P(T) and P(IK), using the same setup as before; in particular, we do not retrain the *Trained probe* and do not adapt the prompts in any way.[8] To test robustness on paraphrases, we generate 10 different paraphrase sets—each holding different formulations of the original inputs—and compute AUPRC on each set. We notice that the AUPRC remains stable for all methods (full results in Appendix C), indicating they are robust to paraphrasing. The most affected method is the *Trained probe* in the P(IK) setting, but even here we only note up to a standard deviation of 3 percentage points (for Mistral-7B-v0).

For translations, we compute a separate AUPRC on the French and Polish versions of T-REx. We find varying degrees of method transferability to new languages. All methods generalize to both French and Polish above chance, except for 1) *Verbalized Confidence*, and 2) *Surrogate Logits* when applied to MistralAI models (see Fig. 7 in Appendix for full results). Notably, the probes trained on English data remain to a large extent reliable (AUPRC for French: .73-.91; for Polish: .61-.91) on unseen languages—with 40B+ models and the instruction-tuned Mistral demonstrating the most transferability. This provides additional evidence for out-of-domain generalization of trained probes (Section 5.3). In particular, the probes can extract scores that are discriminative of true and false facts also from hidden states computed from inputs in a different language than the one used at training. This suggests that the LLMs encode factual confidence in a similar way across languages.

6.2 Robustness of Facts Encoding in LLMs

We hypothesize that, to robustly learn facts and minimize hallucinations, a model has to build stable abstractions over different types of relevant evidence from the training data. We also expect that if the model has built such a robust representation of a fact, this would lead to equal confidence under equivalent formulations of that fact. Inconsistent confidence would in turn indicate excessive reliance on surface-level features.

Fig. 4 shows how paraphrasing the input (8 paraphrase/input) causes changes in $P(\mathrm{T})$ estimates across the Lama T-REx dataset. The amount of variation is not stable across facts: On a large amount

[8]Note this also applies to translations; i.e., the trained probe is trained on English data only and we use English prompts to query the model about French/Polish inputs.

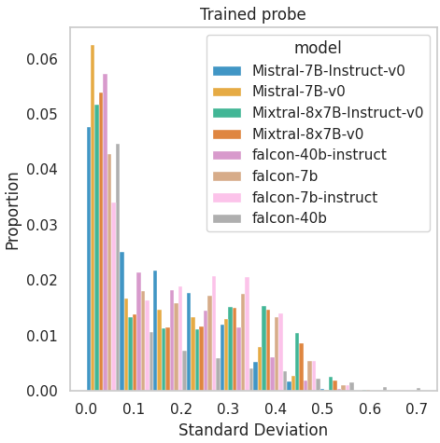

Figure 4: Distribution of standard deviation scores computed on normalized $P(\mathrm{T})$ scores for paraphrases of the same fact.

there is no variation, indicating a stable fact encoding; but on other facts, different wordings lead to varying degrees of confidence, up to .4 standard deviation. This indicates inconsistent LLM behavior with excessive sensitivity to how a claim is worded.

To test robustness of factual knowledge across languages, we compare the distributions of $P(\mathrm{T})$ scores over the same facts using the Spearman correlation analysis (for language pairs) and the Friedman test (for language groups). Analysis reveals high correlations (Spearman's $\rho > .7$; full results in Table 4 in Appendix) between factual confidence scores on all language pairs for the 40B+ models. In particular, we note the highest correlations (in the .87-.92 range) for Falcon 40B models, which points to highly robust multilingual behavior. However, the Friedman tests reveal that for all models, the differences across the distributions are statistically significant (p-values very close to 0); i.e., the differences in scores across the languages are not close enough to be coming from the same population. Given those results, we conclude that while there is a link between the confidence scores across the languages, this is not fully systematic.

7 Discussion & Conclusion

In this paper, we compare existing methods to estimate LLMs factual confidence. Obtaining reliable estimates can benefit LLMs applications, by anticipating potential hallucinations and limiting the non-factual information output by a model (Tonmoy et al., 2024; Evans et al., 2021). However, if not reliable, such estimates can be counterproductive, as they would introduce errors and negatively affect user-model interactions.

Our experiments across eight LLMs demonstrate that *Trained Probe* is the most reliable estimator of LLM factual confidence. It works well for both fact-verification ($P(\mathrm{T})$) and Question Answering ($P(\mathrm{IK})$) consistently across all models, indicating that its reliability is likely to generalize to other LLMs. Unfortunately, applying this method has strong requirements: 1) access to model weights –not always provided for proprietary LLMs, and 2) supervision data. If these requirements cannot be met, but the model is instruction-tuned (Ouyang et al., 2022) we recommend estimating $P(\mathrm{T})$ with *Verbalized Confidence* or *Surrogate Probabilities*. The other methods under study, especially if applied to non-instruction-tuned LLMs, are not consistently reliable.

Our results highlight the need for more research on developing reliable estimators that can be applied to black-box models, whose internal representations cannot be accessed. We expect that the reliability gap of methods like *Verbalized Confidence* with respect to the *Trained Probe* gets smaller with increasingly powerful LLMs, especially in their ability to follow instructions. However, the strong results of *Trained Probe* indicate that hidden states contain signal about factual confidence and it is unclear whether this is fully leveraged by the prompting approaches.

Besides the comparison among methods, we also provide insights on the stability of factual knowledge in LLMs (Petroni et al., 2019; Mahowald et al., 2023; Mitchell and Krakauer, 2023). We show that the factual confidence of an LLM is not always consistent under meaning-preserving variations of the input (paraphrases and translations): while the model may sometimes be sure that a fact is true or false, or that it knows the answer to a question, it may actually behave differently if we reformulate the statement/question. An interesting direction for future research is the exploration of training methods that teach an LLM to better disentangle facts from the diversity of forms they can be stated in, and ultimately exhibit better and more consistent factual knowledge. This would also contribute to increasing LLMs resistance to adversarial attacks (Madry et al., 2018), mitigating the generation of misinformation due to an incorrect sensitivity to input changes.

Limitations

Given the extensive scope of this work (8 models, 5 methods and 2 facets of factual knowledge), we did not have the capacity to study more complex aspects of factual confidence, such as non-atomic facts, reasoning or in-context learning. While our results show that the *Trained probe* is much stronger than other methods on T-REx and PopQA, there is no guarantee that this remains the case in more complex settings. Furthermore, methods themselves have limitations, making comparison use-case dependent. The *Trained probe* method for example requires training data, and while we have tested for transfer capabilities in our simple atomic fact setup, (Kadavath et al., 2022) have shown that there are limits to the kind of tasks this method can be transferred to. The same can be said of the *Sequence probability* method, which in our experiments works better than both prompt-based methods for non instruction fine-tuned models. While this method performs well on simple atomic facts, more complex sentences, or even simple but longer sentences could lead to weaker results. Furthermore, both prompt-based methods are very sensitive to prompt-variations.

Ethics and Broader Impact

This work contributes to the wider goal of automatically reducing risk when using LLMs. We contribute to false fact detection, and answer confidence, leading to potential applications which can build trust in LLMs. None of the methods studied completely solve the issue of hallucination, or non-factual utterances of models, leaving a need for future works on the subject. While methods studied can work with models with 7B and 40B+ parameters, the deployment of those models requires specific infrastructure, and is compute intensive.

References

Ebtesam Almazrouei, Hamza Alobeidli, Abdulaziz Alshamsi, Alessandro Cappelli, Ruxandra Cojocaru, Mérouane Debbah, Étienne Goffinet, Daniel Hesslow, Julien Launay, Quentin Malartic, Daniele Mazzotta, Badreddine Noune, Baptiste Pannier, and Guilherme Penedo. 2023. The falcon series of open language models.

Akari Asai, Zeqiu Wu, Yizhong Wang, Avirup Sil, and Hannaneh Hajishirzi. 2023. Self-rag: Learning to retrieve, generate, and critique through self-reflection.

Amos Azaria and Tom Mitchell. 2023. The internal state of an LLM knows when it's lying.

Yuntao Bai, Andy Jones, Kamal Ndousse, Amanda Askell, Anna Chen, Nova DasSarma, Dawn Drain, Stanislav Fort, Deep Ganguli, Tom Henighan, et al. 2022. Training a helpful and harmless assistant with reinforcement learning from human feedback. *ArXiv preprint*.

Emily M. Bender, Timnit Gebru, Angelina McMillan-Major, and Shmargaret Shmitchell. 2021. On the dangers of stochastic parrots: Can language models be too big? In *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*, FAccT '21.

Emily M. Bender and Alexander Koller. 2020. Climbing towards NLU: On meaning, form, and understanding in the age of data. In *Proc. of ACL*.

Collin Burns, Haotian Ye, Dan Klein, and Jacob Steinhardt. 2022. Discovering latent knowledge in language models without supervision.

Nicholas Carlini, Daphne Ippolito, Matthew Jagielski, Katherine Lee, Florian Tramer, and Chiyuan Zhang. 2022. Quantifying memorization across neural language models. In *The Eleventh International Conference on Learning Representations*.

Yanai Elazar, Nora Kassner, Shauli Ravfogel, Abhilasha Ravichander, Eduard Hovy, Hinrich Schütze, and Yoav Goldberg. 2021. Measuring and improving consistency in pretrained language models. *Transactions of the Association for Computational Linguistics*.

Owain Evans, Owen Cotton-Barratt, Lukas Finnveden, Adam Bales, Avital Balwit, Peter Wills, Luca Righetti, and William Saunders. 2021. Truthful ai: Developing and governing ai that does not lie. *ArXiv preprint*.

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*.

Yarin Gal and Zoubin Ghahramani. 2016. Dropout as a bayesian approximation: Representing model uncertainty in deep learning. In *Proc. of ICML*, JMLR Workshop and Conference Proceedings.

Jiahui Geng, Fengyu Cai, Yuxia Wang, Heinz Koeppl, Preslav Nakov, and Iryna Gurevych. 2023. A survey of language model confidence estimation and calibration.

Chuan Guo, Geoff Pleiss, Yu Sun, and Kilian Q. Weinberger. 2017. On calibration of modern neural networks. In *Proc. of ICML*, Proceedings of Machine Learning Research.

Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Ye Jin Bang, Andrea Madotto, and Pascale Fung. 2023. Survey of hallucination in natural language generation. *ACM Comput. Surv.*, (12).

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2023. Mistral 7b.

Albert Q. Jiang, Alexandre Sablayrolles, Antoine Roux, Arthur Mensch, Blanche Savary, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Emma Bou Hanna, Florian Bressand, Gianna Lengyel, Guillaume Bour, Guillaume Lample, Lélio Renard Lavaud, Lucile Saulnier, Marie-Anne Lachaux, Pierre Stock, Sandeep Subramanian, Sophia Yang, Szymon Antoniak, Teven Le Scao, Théophile Gervet, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2024. Mixtral of experts.

Zhengbao Jiang, Frank F. Xu, Jun Araki, and Graham Neubig. 2020. How can we know what language models know? *Transactions of the Association for Computational Linguistics*.

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

Nora Kassner, Philipp Dufter, and Hinrich Schütze. 2021. Multilingual LAMA: Investigating knowledge in multilingual pretrained language models. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*.

Lorenz Kuhn, Yarin Gal, and Sebastian Farquhar. 2023. Semantic uncertainty: Linguistic invariances for uncertainty estimation in natural language generation.

Moritz Laurer, Wouter Atteveldt, Andreu Casas, and Kasper Welbers. 2023. Less annotating, more classifying: Addressing the data scarcity issue of supervised machine learning with deep transfer learning and bert-nli. *Political Analysis*.

Stephanie Lin, Jacob Hilton, and Owain Evans. 2022a. Teaching models to express their uncertainty in words.

Stephanie Lin, Jacob Hilton, and Owain Evans. 2022b. TruthfulQA: Measuring how models mimic human falsehoods. In *Proc. of ACL*.

Linhao Luo, Trang Vu, Dinh Phung, and Reza Haf. 2023. Systematic assessment of factual knowledge in large language models. In *Findings of the Association for Computational Linguistics: EMNLP 2023*.

Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. 2018. Towards deep learning models resistant to adversarial attacks. In *Proc. of ICLR*.

Kyle Mahowald, Anna A Ivanova, Idan A Blank, Nancy Kanwisher, Joshua B Tenenbaum, and Evelina Fedorenko. 2023. Dissociating language and thought in large language models: a cognitive perspective. *ArXiv preprint*.

Alex Mallen, Akari Asai, Victor Zhong, Rajarshi Das, Daniel Khashabi, and Hannaneh Hajishirzi. 2022. When not to trust language models: Investigating effectiveness of parametric and non-parametric memories.

Potsawee Manakul, Adian Liusie, and Mark J F Gales. 2023. SelfCheckGPT: Zero-resource black-box hallucination detection for generative large language models.

Melanie Mitchell and David C. Krakauer. 2023. The debate over understanding in ai's large language models. *Proceedings of the National Academy of Sciences*, (13).

Xenia Ohmer, Elia Bruni, and Dieuwke Hupkes. 2023. Separating form and meaning: Using self-consistency to quantify task understanding across multiple senses. *CoRR*.

OpenAI. 2023. Gpt-4 technical report. *ArXiv preprint*.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. 2022. Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems*.

Lorenzo Pacchiardi, Alex J Chan, Sören Mindermann, Ilan Moscovitz, Alexa Y Pan, Yarin Gal, Owain Evans, and Jan Brauner. 2023. How to catch an AI liar: Lie detection in black-box LLMs by asking unrelated questions.

Fabio Petroni, Tim Rocktäschel, Sebastian Riedel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, and Alexander Miller. 2019. Language models as knowledge bases? In *Proc. of EMNLP*.

Jirui Qi, Raquel Fernández, and Arianna Bisazza. 2023. Cross-lingual consistency of factual knowledge in multilingual language models. In *Proc. of EMNLP*.

Noam Shazeer, Azalia Mirhoseini, Krzysztof Maziarz, Andy Davis, Quoc V. Le, Geoffrey E. Hinton, and Jeff Dean. 2017. Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. In *Proc. of ICLR*.

Alex Tamkin, Miles Brundage, Jack Clark, and Deep Ganguli. 2021. Understanding the capabilities, limitations, and societal impact of large language models. *ArXiv preprint*.

James Thorne, Andreas Vlachos, Oana Cocarascu, Christos Christodoulopoulos, and Arpit Mittal. 2018. The fact extraction and VERification (FEVER) shared task. In *Proceedings of the First Workshop on Fact Extraction and VERification (FEVER)*.

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.

S. M Towhidul Islam Tonmoy, S M Mehedi Zaman, Vinija Jain, Anku Rani, Vipula Rawte, Aman Chadha, and Amitava Das. 2024. A comprehensive survey of hallucination mitigation techniques in large language models.

Elena Voita, Rico Sennrich, and Ivan Titov. 2019. The bottom-up evolution of representations in the transformer: A study with machine translation and language modeling objectives. In *Proc. of EMNLP*.

Cunxiang Wang, Xiaoze Liu, Yuanhao Yue, Xiangru Tang, Tianhang Zhang, Cheng Jiayang, Yunzhi Yao, Wenyang Gao, Xuming Hu, Zehan Qi, Yidong Wang, Linyi Yang, Jindong Wang, Xing Xie, Zheng Zhang, and Yue Zhang. 2023a. Survey on factuality in large language models: Knowledge, retrieval and domain-specificity.

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

Laura Weidinger, John Mellor, Maribeth Rauh, Conor Griffin, Jonathan Uesato, Po-Sen Huang, Myra Cheng, Mia Glaese, Borja Balle, Atoosa Kasirzadeh, et al. 2021. Ethical and social risks of harm from language models. *ArXiv preprint*.

Miao Xiong, Zhiyuan Hu, Xinyang Lu, Yifei Li, Jie Fu, Junxian He, and Bryan Hooi. 2023a. Can LLMs express their uncertainty? an empirical evaluation of confidence elicitation in LLMs.

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

Zhangyue Yin, Qiushi Sun, Qipeng Guo, Jiawen Wu, Xipeng Qiu, and Xuanjing Huang. 2023. Do large language models know what they don't know?

A Implementation of scoring methods

A.1 Verbalised

In practice: We use the prompt from Tian et al. (2023) with slight variations to adapt to the current task, then filter for non-numerical outputs and normalise.

Prompt used for $P(\text{T})$ experiments: *Provide your confidence level (on a scale of 1.0 to 10.0) that the following statement is correct* **.** `\n\n`*The statement is: $statement* `\n\n`*Confidence level:* Prompt used for $P(\text{IK})$ experiments: *Provide your confidence level (on a scale of 1.0 to 10.0) that you can answer the following question* **.** `\n\n`*The question is: $question* `\n\n`*Confidence level:*

A.2 Surrogate token probabilities

For $P(\text{T})$ we use *"Q: Is the following statement true: $statement (Yes/No/Maybe)?* `\n`*A: "*. For $P(\text{IK})$ we use *"Q: Do you know the answer to the following question: $question (Yes/No/Maybe)?* `\n`*A: "*.

A.2.1 Consistency

We output 10 samples of up to 25 tokens (leaving ample room for the ground truth to be generated in a sentence. Ground truth is generally a word, ranging between two and five tokens) using a temperature of 1, and calculate the final consistency score by computing the mean NLI (Laurer et al., 2023) score of the 10 samples.

B Paraphrasing

B.1 Prompt

Prompt used to generate paraphrases with Mixtral-8x7B-Instruct-v0.1: *Given a sentence, generate paraphrases of it as follows:* `\n\t-` *You can change and/or add words, and/or change the syntactic structure of the sentence;* `\n\t-` *Make sure the new sentence does not add additional details with respect to the original.* `\n\t-` *Make sure the new sentence does not omit any details with respect to the original.* `\n\t-` *Make sure the new sentence is natural and plausible, in spite of the changes.* `\n\t-` *Do not generate the original sentence or previously generated ones.* `\n`*List your paraphrases as bullet-point.* `\n`*Sentence: $sentence* `\n`*New sentences:*

B.2 Paraphrase examples

Original sentence from the *Lama T-REx* dataset: *Michie Mee is a actress by profession* . Paraphrases:

- Acting is the profession of Michie Mee.
- Michie Mee makes a living as an actress.
- Michie Mee is a professional actress.
- Michie Mee is an actress in her profession.
- Michie Mee is an artist who acts for a living.

Original sentence from the *Lama T-REx* dataset: *The Munsters was originally aired on Bravo network* . Paraphrases:

- Bravo network was the first to air The Munsters.
- The Munsters was first shown on Bravo.
- The Munsters was first transmitted on Bravo.
- Bravo was the first network to air The Munsters.
- The original network that broadcast The Munsters was Bravo.
- The Munsters was first broadcasted on Bravo.

Original sentence from the *PopQA* dataset: *What is George Rankin's occupation?* Paraphrases:

- What does George Rankin do for a living?
- What line of work is George Rankin in?
- What is George Rankin's job?
- What is George Rankin's profession?
- Can you tell me what George Rankin does?
- George Rankin's employment, could you tell me about it?
- George Rankin's work, what is it?

Original sentence from the *PopQA* dataset: *In what city was Louis Renault born?* Paraphrases:

- Where did Louis Renault come into the world?
- In which urban area did Louis Renault enter the world?
- In what metropolis did Louis Renault make his appearance?
- In which city did Louis Renault first see the light of day?

- In which city was Louis Renault given birth?
- In what city was Louis Renault brought into the world?
- In what city was Louis Renault born into the world?

C Method robustness to variation

In Figure 5 and 6 we randomly sample a paraphrase for every sentence in the original dataset, making ten sets of paraphrases of the same size. We then compute AUPRC without changing the method in any way for the ten sets, and look at the variation. All methods remain stable, and robust to paraphrases. The biggest variation occurs for the *Trained probe method*, but are only of the order of 3 percentage points. Table 4 shows the correlation between scores across different languages, and Figure 7 shows AUPRC of all 4 methods for French and Polish Lama T-RE.

Name	Size	En-Fr	En-Po
Falcon	40B	.90	.86
Falcon Ins.	40B	.92	.87
Falcon	7B	.79	.44
Falcon Ins	7B	.67	.35
Mistral	7B	.67	.58
Mistral Ins	7B	.65	.53
Mixtral	46.7B	.87	.77

Table 4: Spearman correlation coefficient for English-French and English-Polish $P(T)$ scores on translated Lama T-REx statements.

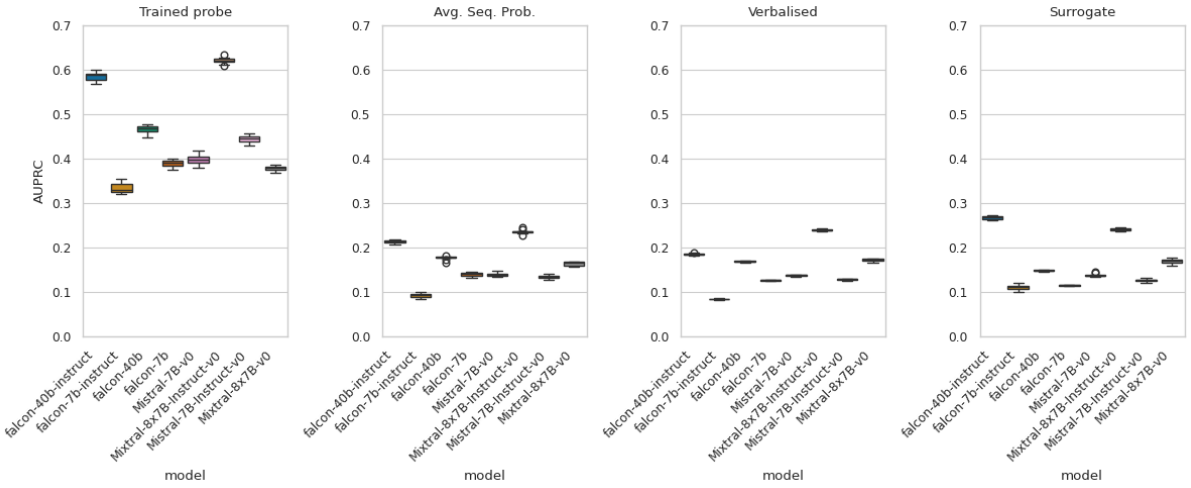

Figure 5: Variation in $P(\text{IK})$ AUPRC when sampling paraphrases. 10 sets of paraphrases are randomly sampled, with one paraphrase for every question in PopQA.

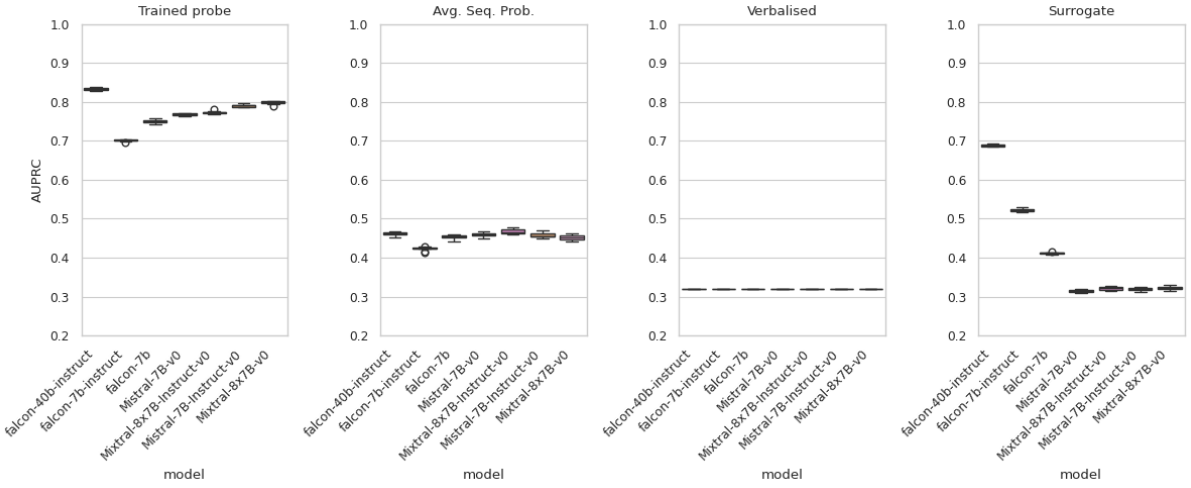

Figure 6: Variation in $P(\text{T})$ AUPRC when sampling paraphrases. 10 sets of paraphrases are randomly sampled, with one paraphrase for every question in Lama Lama T-RE.

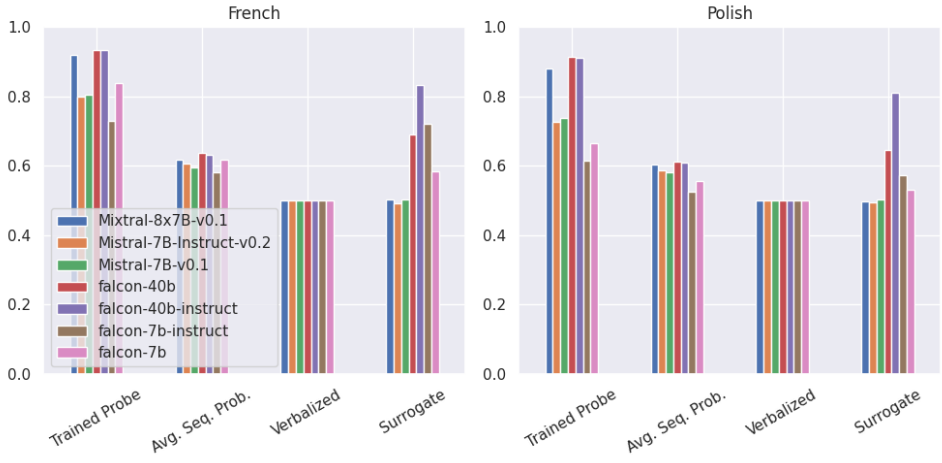

Figure 7: AUPRC for $P(T)$ scores for translations in French and Polish of the Lama T-REx statements.