Calibration of large language models based on the conformal prediction

Shamil Chankaev, Eugene Ilyushin

Abstract: **Modern language models are widely used across various domains due to their ability to generate coherent and contextually relevant text. However, despite significant advancements, such models remain prone to errors and may produce hallucinations—statements that are well-formed but factually incorrect. This article proposes a novel approach to evaluating uncertainty in language model responses based on conformal prediction methods. It explores techniques for calibrating the models' probabilistic estimates in order to enhance the reliability and interpretability of their outputs. The study demonstrates how the proposed method can more effectively identify potential errors and improve the justification of generated responses. The results open up new possibilities for increasing the reliability of language models in critical applications where accuracy and confidence in responses are of paramount importance.**

key words: **conformal prediction, model calibration, uncertainty quantification function, large language models**

I. Introduction

In recent years, language models (LMs) such as GPT and BERT have demonstrated significant progress in Natural Language Processing (NLP) tasks. They are successfully applied in diverse scenarios—from machine translation and text summarization to code generation. However, despite these impressive achievements, several unresolved issues remain, among which the problem of calibration—the model's ability to adequately assess its confidence in its own predictions—is particularly important.

The lack of reliable confidence assessment limits the application of LMs in safety-critical domains where the cost of error can be high. In fields such as medicine, law, and finance, not only accuracy is required, but also the ability to identify situations where the model might err. The inability to do so increases the risk of disseminating incorrect information and undermines trust in such systems.

Conformal prediction methods represent a promising approach to addressing this challenge. These methods allow for the generation of statistically sound predictions with a controlled level of reliability. This work explores the application of the conformal approach for calibrating language models. Particular attention is given to the development of a novel uncertainty quantification method based on the analysis of token distributions in generated responses.

This paper presents an experimental investigation into the effectiveness of the proposed method, along with a comparison to existing approaches. The obtained results demonstrate the potential of conformal prediction as a tool for enhancing the reliability and interpretability of language models.

II. Related Work

- **Learning by transduction** - Vapnik V., Vovk V., Gammerman, 2013 [1]. In this work, the authors investigated the applicability of conformal prediction methods to heteroscedastic and sparse data. The method proposed in this work formed the basis of transductive learning -— a conformal prediction method applicable to Natural Language Processing (NLP) tasks.
- **Least ambiguous set-valued classifiers (LAC)** - Sadinle et al., 2019 [2]. In this paper, the authors focused on applying conformal prediction to classification tasks with a large number of classes. Their developed Least Ambiguous set-valued Classifier (LAC) function allowed for effective uncertainty estimation in model responses and became widely used for model calibration in classification tasks, including the evaluation of large language models.
- **Conformal Prediction with Large Language Models for Multi-Choice Question Answering** - Kumar et al., 2023 [3]. In this article, the authors investigated the applicability of conformal prediction methods to modern LLMs and the improvement of their accuracy. Building on previous work, they succeeded in enhancing the quality of model calibration. However, the method for calibrating language models proposed in this article is not universal – an uncertainty scoring function needs to be developed separately for each task. Also, it was found that the quality of model calibration strongly depends on the prompt -— the formulation of the question.
- **Robots That Ask For Help: Uncertainty Alignment for Large Language Model Planners** - Allen Z. Ren, Anushri Dixit et al., 2023 [4]. In this paper, the authors applied conformal prediction to identify model uncertainty in tasks involving the model's interaction with the environment and the execution of commands in natural language. In their work, the authors highlighted the main difficulties arising in solving these tasks—the ambiguity of natural language, problems in understanding language commands, etc. To address these problems, it was necessary to identify moments of model uncertainty and send a clarifying query to obtain additional information. To identify these moments, the researchers created the KNOWNO framework, which allowed for the detection of model uncertainty and the clarification of user queries.

- **Scalable Best-of-N Selection for Large Language Models via Self-Certainty** - Zhewei Kang, Xuandong Zhao, Dawn Song, 2025 (Note: Year might be a placeholder or future publication, check if this is intended) [5]. In this article, the authors investigated a method for assessing the uncertainty of language models directly through the analysis of the probability distribution for each token in the generated response. The uncertainty estimation methods they proposed allowed for the calibration of language models for any task.

III. Principles of Language Model Operation

Modern language models, including architectures from the GPT family, are typically implemented as autoregressive transformers. The fundamental principle of their operation involves the sequential generation of an output sequence, where each token is predicted conditionally based on the input query and previously generated tokens. This factorization of the probability space allows the task of text generation to be formalized as a sequential prediction problem.

Let q be the input query, and $a = (a1, a2, \ldots, an)$ be the output text sequence composed of tokens. Then, the conditional probability of generating the complete sequence a can be represented as follows:

$$p(a|q) = \prod_{i=1}^{n} p(a_i|q, a_{<i})$$

where $a_{<i} = (a_1, \ldots, a_{i-1})$ are the preceding tokens.

At each step i, the model generates a probability distribution $p(a_i \square q, a_{<i})$ over the vocabulary of possible tokens, reflecting the estimated probability of each token appearing in that position. A specific token is selected either by sampling from this distribution or by deterministically choosing the most probable token (e.g., using greedy decoding or beam search).

Such an autoregressive structure enables models to effectively consider context and construct coherent text outputs. However, it also imposes limitations: errors in early steps can accumulate, and the model itself lacks an inherent mechanism for globally controlling the reliability of the entire sequence. Consequently, models may ”hallucinate,” especially when errors occur in the initial stages of generation. This makes the task of reliable uncertainty estimation and subsequent calibration of output distributions particularly crucial when applying language models in high-stakes domains.

IV. Conformal Prediction Method

Conformal prediction is a powerful tool for constructing prediction intervals and for model calibration. These intervals are constructed using an uncertainty quantification function for the model's responses. By analyzing the behavior of the uncertainty quantification function, we can detect hallucinations based on the model's anomalous responses. Consequently, it becomes possible to calibrate the model to improve the quality of its responses [6].

A. Uncertainty Scoring Function

More formally, the uncertainty scoring function can be defined as follows:

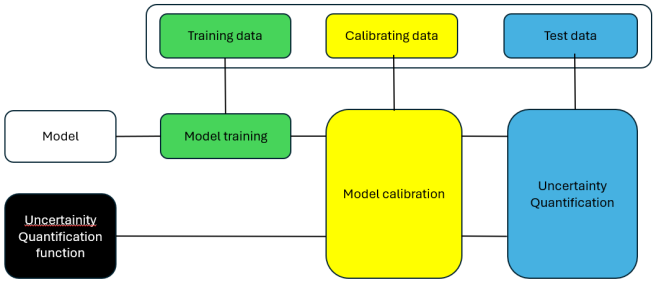


Figure 1. Model calibrating scheme

$$F : R^m \to [0, 1]$$

Where R^m is the space of model responses, and $[0, 1]$ represents the model's uncertainty, expressed numerically. Accordingly, if $F(a_1) < F(a_2)$, then the model is more confident in response a_1 than in a_2. And if the difference between the values becomes too large, this may indicate the detection of an anomaly. In Natural Language Processing (NLP) tasks, particularly for multiple-choice tasks (or tasks involving the selection of a correct answer), the LAC (Least Ambiguous set-valued Classifier) function is often used:

$$S(X, Y) = 1 - |F(X)|_Y$$

where $|F(X)|_Y$ is the softmax probability of class Y for input X.

A key feature of uncertainty scoring functions is their sensitivity, i.e., their ability to detect anomalies. The more accurately the function detects deviations in the model's responses, the more effectively its responses can be improved.

B. Model Calibration

After selecting the uncertainty scoring function, the model needs to be calibrated. The calibration process proceeds as follows:

- A dataset is collected on which the calibration will be performed. This is typically a small subset of the data on which the model was trained.
- This data is processed by the model, and the obtained results are analyzed using the uncertainty scoring function. Patterns in the model's responses, its typical confidence level, and areas where it is least confident are identified.
- For new data from the test set, prediction intervals are constructed, or the model's confidence level in its predictions is displayed.

It is important to note that for text generation tasks, it is often impossible to construct prediction intervals. Therefore, for the model's generated responses, the degree of confidence in its prediction is most often calculated.

C. Model Response Adjustment

After calibrating the language model, it becomes possible to quantitatively assess its degree of uncertainty during response generation. The resulting probabilistic characteristics of the output distribution allow for an analysis of

how confidently the model makes a specific prediction in a given context. This information can be used for post-processing generation results, in particular—for adjusting or filtering responses that exhibit a high degree of uncertainty. One effective approach in this direction involves constructing prediction intervals for the model's predictions, which allows for the generation of estimates with a controlled level of reliability. Thus, a calibrated model not only provides probabilistic predictions but also accompanies them with additional information that can serve as a basis for decision-making under uncertainty. A prediction interval is constructed as follows:

$$q_\alpha = Quantile(S_{1..n}, \frac{(n+1)(1-\alpha)}{n})$$

One of the key challenges in applying the conformal prediction method to language models is the limited effectiveness of existing uncertainty scoring functions. This is due to the specifics of Natural Language Processing tasks, particularly phenomena such as semantic ambiguity and contextual variability. An additional complexity is the high dimensionality and sparsity of the input space: language data are represented as tokens from an extremely large vocabulary, which complicates the construction of reliable statistical estimates.

These factors significantly limit the application of classical conformal prediction approaches, whose effectiveness largely depends on the stability and representativeness of the features used. Consequently, there is a need to develop specialized methods adapted to the structure of language models' output distributions. This work proposes a novel approach to uncertainty quantification, based on the analysis of the token probability distribution during response generation. The method aims to enhance the sensitivity of the assessment to the linguistic characteristics of the model and to ensure more accurate calibration of output predictions under conditions of high ambiguity.

V. Quantification of Model Confidence

The new method for model uncertainty quantification is based on the following improvements: a weighted assessment of model confidence during the generation of each token, taking into account its importance in the output response, and an analysis of the probability distribution during the generation of each token. These aspects will be elaborated upon below.

A. Weighted Assessment of Model Confidence During the Generation of Each Token

The primary method for assessing model confidence in their predictions is the normalized log probability of the prediction [7]:

$$AvgLogP = -\frac{1}{n}\sum_{i=1}^{n} \log p(a_i|q, a_{<i})$$

Besides its obvious advantages, such as analyzing model confidence across all tokens of the generated response and good interpretability of the results, this metric also has several drawbacks, the most significant of which is low specificity, especially for long sequences. Since natural language is highly redundant, texts generated in natural language often contain prepositions, linking words, and idiomatic expressions, which improve the quality of the response itself but significantly skew the score. This can be seen in the following example. In this example, the Qwen-2.5-1.5B model was used.

Question: If you write out all the numbers from 1 to 1000 in a single line, how many digits will be in this line?

Answer 2: In the sequence of numbers from 1 to 1000, there will be 9 single-digit numbers, 90 two-digit numbers, and 900 three-digit numbers, as well as the four-digit number 1000. Therefore, in our answer, we will have 9*1 + 90*2 + 900*3 + 4 = 2890. Answer: 2890 digits.

In this case, the language model made a calculation error. However, because many tokens in the response were generated with high model confidence (i.e., $p(a_i|q, a_{<i})$ was high for many a_i), the resulting AvgLogP score was low (indicating high overall confidence). This is one of the problems with this metric: when generating long texts, for most tokens $p(a_i|q, a_{<i}) \rightarrow 1$, causing the model's confidence in the response to become high, even if the final answer is incorrect.

This becomes even more problematic given that modern reasoning models can generate very long text chains, making the AvgLogP score uninformative.

Currently, the Best-of-N method is used to address this problem, where multiple responses are generated, and a weighted average of the model's confidence across all responses is taken [8][9]. This method (Best-of-N) demonstrates improved characteristics in quantifying the uncertainty of language models, especially in tasks requiring logical inference and step-by-step reasoning – such as physics, mathematics, and algorithmic problems. In these cases, the accuracy of identifying uncertain predictions significantly increases, contributing to more reliable model calibration. However, for tasks primarily testing factual or terminological knowledge (e.g., in geography, biology, astronomy), the quality improvement is less pronounced. This is because in such tasks, uncertainty is often related not to the inference process but to the absence of relevant information in the model's parameters.

It should also be noted that this approach (Best-of-N) has high computational complexity. Its implementation requires generating multiple responses for the same input query, which significantly increases time and resource costs compared to basic methods.

To increase the sensitivity of uncertainty quantification, it was proposed to modify the aggregation function by amplifying the contribution of tokens with the highest prediction entropy. This approach is based on the empirical observation that errors in the generation of language model responses often occur in segments where the probability distribution is most unstable, and the model exhibits a high degree of uncertainty. It is precisely during the generation of such "weak" tokens that deviations from the correct logical or semantic line of reasoning often occur, ultimately leading to the formation of an incorrect or distorted response.

Thus, assigning increased weight to tokens with high uncertainty allows for a more accurate reflection of the model's true confidence in the correctness of the complete response. It is assumed that an increase in the proportion of such tokens in the generated sequence directly correlates with

an increased probability of error and can therefore serve as a reliable marker of potential output unreliability. The resulting metric takes the following form:

$$AvgLogP = -\frac{1}{n}\sum_{i=1}^{n}\frac{\log p(a_i)}{p(a_i)}$$

The use of these weights has made the metric more specific, better at identifying the model's uncertainty in its response.

B. Using Information from Token Distributions

As is known, at each step, language models generate a probability distribution for the next token a_i. Consequently, we can use information from this probability distribution to quantify the model's uncertainty for this token. It can be assumed that the model's maximum uncertainty will manifest as a uniform probability distribution. Therefore, the less equiprobable the distribution, the higher the model's confidence in its response.

To measure the distance between two distributions, the Kullback-Leibler (KL) divergence can be used:

$$KL(P||Q) = \sum_x P(x)\log\frac{P(x)}{Q(x)}$$

In this case, the resulting token uncertainty scoring function will take the following form:

$$KL(U||p) = -\frac{1}{N}\sum_x \log(Np(x))$$

Here, N is the size of the token vocabulary over which the distribution is defined.

This function is more flexible than the classical token generation probability; it allows for the incorporation of information from the generated distribution, thereby enabling more effective identification of the model's uncertainty in its prediction.

Consequently, the final uncertainty scoring function will be as follows:

$$F = -\frac{1}{n}\sum_{i=1}^{n}\frac{KL(U||p_i(\cdot))}{p(a_i)} = \frac{1}{nN}\sum_{i=1}^{n}\frac{\sum_{j=1}^{N}\log(Np_i(x_j))}{p(a_i)}$$

where $x_1, \ldots, x_N$ is the vocabulary of tokens in the generated distribution, and $p_i(x_j)$ is the probability of generating token x_j at the i-th step. The term $p_i(\cdot)$ in $KL(U||p_i(\cdot))$ refers to the probability distribution generated by the model at step i for the next token, and $p(a_i)$ refers to the probability assigned by that distribution to the actually generated token a_i.

VI. Experimental Setup

The primary method for evaluating model quality via conformal prediction techniques is model calibration. For a well-calibrated model, its confidence function regarding its responses should correlate with the correctness of those predicted responses. More formally, let $F(x)$ be the model's confidence score for a response x. Let X_y be the set of all model responses for which the confidence score $F(x)$ is greater than y:

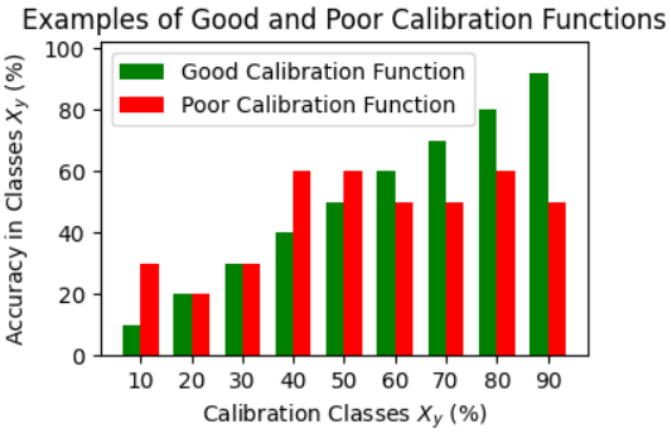


Figure 2. Good and bad calibrating models

$$X_y = \{x : F(x) > y\}$$

Then, a well-calibrated model will satisfy the condition:

$$\frac{|TP(X_y)|}{|X_y|} \geq y$$

That is, the proportion of correct responses in the set X_y (i.e., among those responses where the model's confidence was greater than y) will be no less than y.

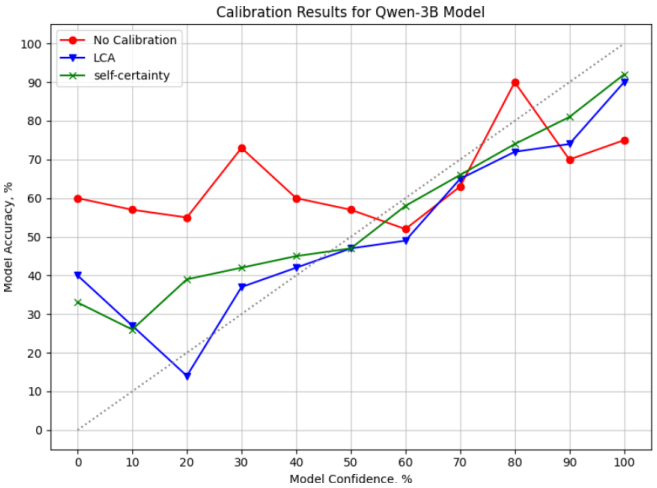
An example of well-calibrated and poorly-calibrated functions is shown in Fig. 2.

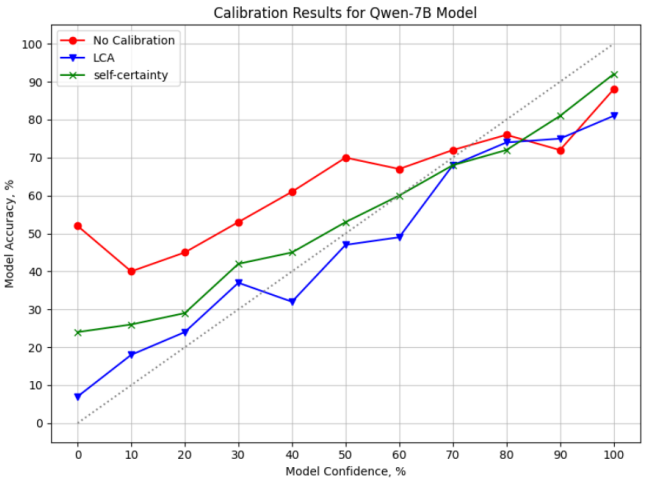
To evaluate the effectiveness of the applied solutions, experiments were conducted on small language models from the Qwen and Llama families. A comparative analysis was performed on the responses of:

1) models without conformal calibration (baseline)
2) models calibrated using the LAC function
3) models calibrated with the method proposed in this article.

Model testing was conducted on the MMLU benchmark. MMLU is a large collection of test questions on various topics used to evaluate the factual knowledge of language models. For testing, only multiple-choice questions where the model had to select one of the provided answer options were used.

To assess the calibration of the baseline models (those not subjected to conformal prediction techniques), their accuracy on the full set of questions was evaluated against their native confidence scores (e.g., the probability assigned to the chosen answer). When evaluating models calibrated with LAC, only the model's final selected answer was analyzed, and its reasoning steps (if any were generated before the final choice) were disregarded for the LAC score calculation. For evaluating models with our proposed method, the model's confidence over the entire generated response (which may include reasoning leading to the final answer) was considered.





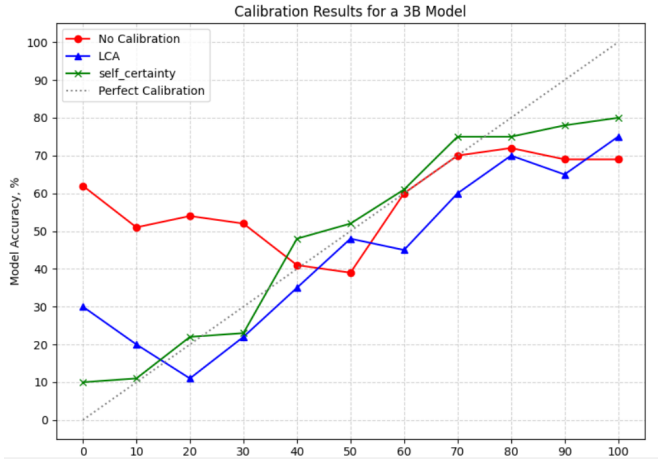


Figure 3. 3B parameters models

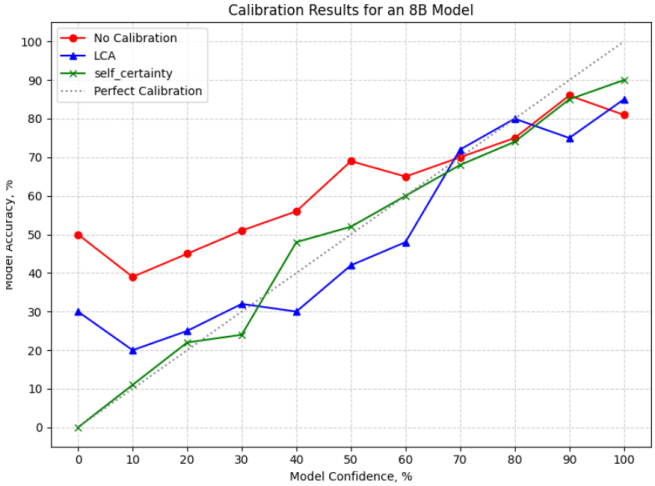


Figure 4. 7-8B parameters models

The results presented in the figures show that the proposed method significantly improved the quality of model responses and ensured more accurate calibration of its predictions. This is particularly noticeable for small and ultra-small models, which have a limited generalization capacity and perform worse on tasks requiring precise knowledge reproduction.

The greatest improvement was achieved in sections testing factual knowledge—geography, anatomy, biology, and astronomy. In these domains, the method effectively identified errors through accurate uncertainty quantification and subsequent response correction.

Furthermore, in tasks involving step-by-step reasoning (mathematics, physics), the new uncertainty scoring function outperformed the standard LCA metric. Its sensitivity to intermediate tokens with high entropy contributed to improved accuracy of the final predictions.

Models with a larger number of parameters possess a higher memorization capacity, which enables them to better reproduce factual knowledge compared to smaller models. Large language models exhibit a higher degree of calibration even without the application of additional methods, including conformal prediction.

When comparing the Qwen and LLaMA architectures, no significant differences in response quality were observed—their predictions were comparable in accuracy.

The uncertainty scoring function proposed in this work demonstrated particularly high effectiveness in physics and mathematics tasks, which are characterized by detailed and step-by-step responses. The assessment of the probability distribution at the individual token level allowed for the identification of segments with high uncertainty, thereby enhancing the reliability of predictions.

At the same time, more modest results were obtained in humanities disciplines—history, law, archaeology, and sociology. Here, high response accuracy requires access to specific facts that smaller models often lack. However, as the number of parameters increases, the model demonstrates improvement in these tasks, which indicates its capacity for memorizing and generalizing factual information from a wide range of domains.

References

[1] Gammerman Alex, Vovk Volodya, Vapnik Vladimir. Learning by transduction // arXiv preprint arXiv:1301.7375. — 2013. — URL: https://arxiv.org/abs/1301.7375.

[2] Sadinle Mauricio, Lei Jing, Wasserman Larry. Least ambiguous set-valued classifiers with bounded error levels // Journal of the American Statistical Association. — 2019. — Vol. 114, no. 525. — P. 223–234. — URL: https://doi.org/10.1080/01621459.2017.1395341.

[3] Conformal prediction with large language models for multi-choice question answering / Bhawesh Kumar, Charlie Lu, Gauri Gupta et al. // arXiv preprint arXiv:2305.18404. — 2023. — URL: https://arxiv.org/abs/2305.18404.

[4] Robots that ask for help: Uncertainty alignment for large language model planners / Allen Z. Ren, Anushri Dixit, Alexandra Bodrova et al. // arXiv preprint arXiv:2307.01928. — 2023. — URL: https://arxiv.org/abs/2307.01928.

[5] Kang Zhewei, Zhao Xuandong, Song Dawn. Scalable best-of-n selection for large language models via self-certainty // arXiv preprint arXiv:2502.18581. — 2025. — URL: https://arxiv.org/abs/2502.18581.

[6] Angelopoulos Anastasios N., Bates Stephen. A gentle introduction to conformal prediction and distribution-free uncertainty quantification // arXiv preprint arXiv:2107.07511. — 2021. — URL: https://arxiv.org/abs/2107.07511.

[7] Self-consistency improves chain of thought reasoning in language models / Xuezhi Wang, Jason Wei, Dale Schuurmans et al. // arXiv preprint arXiv:2203.11171. — 2022. — Published at ICLR 2023. URL: https://arxiv.org/abs/2203.11171.

[8] Let's verify step by step / Hunter Lightman, Vineet Kosaraju, Yura Burda et al. // arXiv preprint arXiv:2305.20050. — 2023. — URL: https://arxiv.org/abs/2305.20050.

[9] Math-shepherd: Verify and reinforce llms step-by-step without human annotations / Peiyi Wang, Lei Li, Zhihong Shao et al. // Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers). — Bangkok, Thailand : Association for Computational Linguistics, 2024. — August. — P. 9426–9439. — URL: https://aclanthology.org/2024.acl-long.510.