A Model API/Checkpoints

This section provides a pointer to checkpoints that we used during experiment. All open-source models are available on the Hugging Face platform. For LLaMA2, we use "meta-llama/Llama-2-(7, 13, 70)b-chat-hf" respectively. For Mixtral MOE, we use "mistralai/Mixtral-8x7B-Instruct-v0.1". For DeepSeekMoE, we use "deepseek-ai/deepseek-moe-16b-chat". For InstructScore, we use "xu1998hz/InstructScore". For the translation model Madlad400-10b, we use "google/madlad400-10b-mt". We used GPT-3.5-Turbo and GPT-4 from OpenAI platform (https://platform.openai.com). We use gemini-pro from Google Gemini API.

B Quantile Mapping

While BLEURT (Sellam et al., 2020) correlates highly with human judgments (Freitag et al., 2022), its scale of roughly 0 to 1 is incompatible with the MQM human annotations, which range from -25 to 0. A linear mapping is not feasible, as the BLEURT score is not calibrated to the human score, meaning a BLEURT score of 0.8 does not correspond to -5 in MQM annotations.

To address this issue, we employ quantile mapping (Cannon et al., 2015) to transform the BLEURT score into the distribution of human scores. This method involves learning a mapping function that maps the quantiles or percentiles of the predictive distribution to those of the observed distribution. In this case, our predictive distribution is derived from the BLEURT score distribution, while our observed distribution comes from the corresponding human score distribution.

We utilize the WMT22 shared metric task (Freitag et al., 2022) to obtain mapped BLEURT-human scoring pairs. In this shared metric task, each translation generated by different translation model is rated by humans using the MQM human rating scale. We also run BLEURT on the same set of translations to obtain BLEURT scores, resulting in 28125 mapped BLEURT-human scoring pairs.

We then perform the following steps: 1) Separately sort the data of the two distributions in ascending order. 2) Compute the cumulative distribution function (CDF) for each distribution. 3) Learn an interpolation function that maps the percentiles of the first distribution to the percentiles of the second distribution. 4) Apply the mapping function to the values drawn from the predictive distribution (BLEURT score distribution) to obtain the corresponding values in the observed distribution (human MQM score distribution).

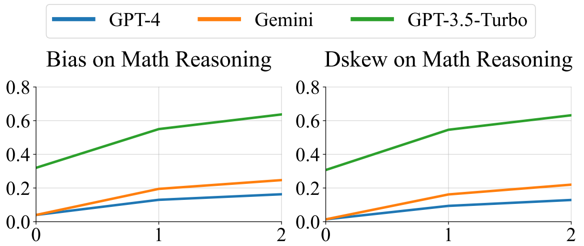

Figure 11: Bias and distance skewness in generated texts from GPT-4, GPT-3.5-Turbo, and Gemini are measured on MATH testing set throughout the self-refinement steps. Results show an increase in bias and skewness during iterative self-consistency, causing biased ensembles in reasoning paths.

This process maps the BLEURT score distribution to the human score distribution (from -25 to 0) while preserving the relative ordering of BLEURT scores. In our experiments, we used the latest BLEURT model, BLEURT-20 checkpoint (Pu et al., 2021), which demonstrates the highest correlation to the human judgments among its variants.

C Gemini's Skewness at Translation

Specifically, in the Java-English (Jav-En) language pair, Gemini initially assigns lower quality scores to its output compared to BLEURT assessments during early iterations, resulting in an underestimation of output performance. This phenomenon accounts for the decrease in distance skewness at the beginning, as the right-skewed distribution becomes more neutral. However, as bias accumulates in later iterations, the distribution shifts towards a left-skewed distribution, leading to an increase in distance skewness.

D Self-consistency results on Math reasoning

We slightly modify the self-refine pipeline by replacing the self-evaluation with self-consistency verification (Huang et al., 2023a). Namely, with the initial solution, LLM will generate an additional ten reasoning paths and a majority vote for a proposed answer. If the proposed answer is inconsistent with the prior solution, we will output a binary score of 0, and the initial answer will be replaced by the proposed answer. Otherwise, we will output a score of 1, and no change will be made to the initial answer. Figure 11 illustrates that all large language models (LLMs) exhibit an increase in bias and skewness estimation in the iterative self-consistency pipeline. This suggests that LLMs introduce self-biases towards certain reasoning paths during self-refine, ultimately leading to a biased ensemble across multiple reasoning paths.

E Additional Results

In Table 5, we include human evaluation results and GPT-4's quality scores for the 0th and 10th iteration of refinement generation at Yorba-to-English. In Table 6, we include human evaluation and GPT-3.5-Turbo's quality assessment on the 0th and 10th iteration of refinement generation at Yorba-to-English. In Table 7, we include human evaluation and Gemini's quality assessment on the 0th and 10th iterations of refinement generation. In Figure 12, we include full bias and distance skewness for Yor-En, Jav-En, Arm-En and Ig-En translations on Flores200.

Human Evaluation	Human	GPT-4	Bias	Dskew
0th Iteration	-15.0	-6.92	8.06	0.452
10th Iteration	-15.1	-0.52	14.6	0.692

Table 5: This table presents human evaluation results and GPT-4's quality scores for the 0th and 10th iteration of refinement generation performed at Yor-En. Bias and Dskew estimates are included to quantify the biases identified through human evaluation.

Human Evaluation	Human	GPT-3.5	Bias	Dskew
0th Iteration	-22.2	-2.61	19.6	0.803
10th Iteration	-21.9	-0.03	21.9	0.885

Table 6: We report human evaluation and GPT-3.5-Turbo's quality assessment on the 0th and 10th iteration of refinement generation at Yor-En.

Human Evaluation	Human	Gemini	Bias	Dskew
0th Iteration	-17.3	-8.92	9.62	0.355
10th Iteration	-18.3	-0.72	17.6	0.766

Table 7: We report human evaluation and Gemini's quality assessment on the 0th and 10th iterations of refinement generation at Yor-En.

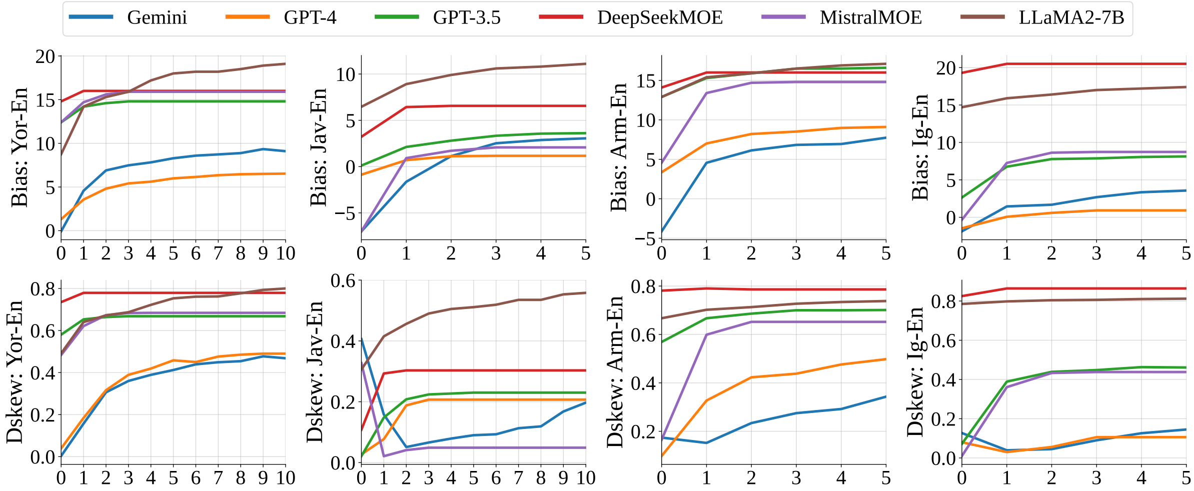

Figure 12: Full Bias and Dskew estimations for Yor-En, Jav-En, Arm-En, and Ig-En translations on FLores200, with the x-axis showing self-refine steps, reveal that all LLMs exhibit self-bias, where open-source LLMs exhibit higher levels than GPT-4 and Gemini.

In-context-learning prompt for LLM's initial generation at translation:
Below is an instruction that describes a task.
Instruction:
Translate Chinese text into English.
Chinese: 新华时评：把优秀返乡农民工打造成乡村振兴生力军-新华网
English: Xinhua Commentary: Outstanding returning rural migrant workers can be a rural revitalization army - Xinhuanet

Below is an instruction that describes a task.
Instruction:
Translate English text into German.
English: You can come back any time as our chat service window is open 24/7
German: Sie können jederzeit wiederkommen, da unser Chat-Service-Fenster täglich rund um die Uhr geöffnet ist

Below is an instruction that describes a task.
Instruction:
Translate Yorba text into English.
Yorba: Won da Olori Skwodroni. Dilokrit Pattavee gege bi awako ofururu.
English: The pilot was identified as Squadron Leader Dilokrit Pattavee.

Below is an instruction that describes a task.
Instruction:
Translate Yoruba text into English.
Yorba: O ko ago ilekun WiFi, O wi.

Output for translation:
Can you please turn off the WiFi, I'm done.

Table 8: Those are the translation in context learning example we used to prompt all LLMs across four language directions at Flores200. In this example, the source translation is Yourba text "O ko ago ilekun WiFi, O wi.". The English output text is from LLaMA2-7B's generation "Can you please turn off the WiFi, I'm done.".