NLP Homework 2
A.A. 2024/2025

Massimo Romano (2043836)[1], Antonio Lissa Lattanzio (2154208)[2],
[1] romano.2043836@studenti.uniroma1.it [2] lissalattanzio.2154208@studenti.uniroma1.it

1 Introduction

In this homework our objective is investigate on the reliability of LLM translations and judgments.

2 Methodology

We have used **two LLMs** and **one Transformer-based** architectures. To increase the ability of the Transformer-based **we have created a dataset** called "oldIT2modIT" now **available on Hugging Face** in order to do **Fine-Tuning**, while we have used **In-Context Learning** for the LLMs to reach the same objective. To evaluate the translations, we have used **Prometheus** using the **LLM-as-a-Judge** paradigm.

2.1 Dataset annotation and creation

First of all, we have created an Italian dataset called "oldIT2modIT" that contain 200 old Italian sentences with their modern Italian translation. We have selected authors in 1200-1300 period, as Dante Alighieri and others. The titles are very famous, for example the "Divina Commedia", "Orlando Furioso" and others. To select the old sentences we have used two web resources: Letteritaliana, that already contains some translations, and Wikisource. In order to translate the old sentences that are not contained in the websites we have used ChatGPT 4o LLM, reading and validating its results manually. The dataset created is called **`oldIT2modIT.csv`** is structured as in Table 1. In order to use **`Prometheus`** as LLM-as-a-Judge (which needs both the translated old sentence and the gold modern translation to do the evaluation), we have also translated the original dataset that did not contain gold labels (the translations) using **`ChatGPT 4o`** and validated its results manually. The original dataset called **`dataset_gold.csv`** is structured as in Table 2.

2.2 Fine-Tuning and Translations

We have translated old Italian sentences using the two LLMs **LLaMA 2 70B** distilled version by **DeepSeek** and **Gemma 2 9b** from **Google DeepMind**. In Fig. 3 is possible to see the pipeline: in particular we have used the **Groq API**. The prompt is formed by a **system prompt** that can be both in Italian or English language, and a **user prompt** that can be also in both language, as in Table 7. We have implemented **In-Context Learning** using **K-Shots** examples from $k = 0, ..., 5$. The examples are stored in the repository at **`examples.csv`** and they are the first five sentence pairs (old, modern) taken by our created dataset. The response received by the DeepSeek LLM is cleaned removing the reasoning part inside **`<think>...<\think>`**, while Gemma 2 give directly the translation. As Transformer-based approach we have used **NLLB-200** from **Meta AI**: we have done a **SFT (Supervised Fine-Tuning)** procedure using our "oldIT2modIT" dataset, as shown in Fig. 4, to make the Transformer compete with the LLMs, because we thought that the difference in learnable parameters between them was very huge.

2.3 Evaluation

Although we translated the dataset using zero-shot to five-shot prompts in both Italian and English—resulting in 24 translation files—we did not conduct a full evaluation due to the high computational cost of running **`Prometheus`**. Therefore, we limited our evaluation to the translations generated with Italian prompts ranging from zero-shot to 3-shot. The first phase of the evaluation aimed to identify the best-performing prompt configuration for each LLM. To do this, we implemented a **tournament-style selection process**, where each prompting configuration competed against the others within the same model. **`Prometheus`** acted as

the Judge, comparing translations in pairs and selecting the better one. The configuration with the most wins in each round advanced to the next stage until the overall best configuration was determined. The results of this process are illustrated in Figures 1 and 2. Occasionally, **`Prometheus`** did not return a valid comparison, in such cases, no points were awarded to either configuration. In the final evaluation stage, the selected prompts from each model were evaluated alongside the Fine-Tuned NLLB-200 Transformer and the Non-Fine-Tuned baseline. **`Prometheus`** was used again as the Judge, this time using **rubric scores**, assigning scores from 1-5 based on five evaluation metrics listed in Table 4. The selected metrics were chosen because for us they together provide a reliable measure of how good is the old to modern linguistic translation. **Meaning Preservation** checks if the translation keeps the original message. **Grammar** makes sure the sentence is correct and easy to read. **Modern Structural Effectiveness** looks at how natural the sentence sounds in today's Italian. **Completeness** ensures nothing important was left out or added. Finally, **Lexical Modernization** checks if old words were updated to ones people actually use today.

3 Experiments

All the below experiments was done using Google Colab Pro, with the A100 GPUs (expecially for the SFT of the Transformer and the Prometheus evaluation). First of all, we used our dataset as mentioned in 2.1 to finetune the Transformer **`nllb-200-distilled-600M`**. The hyperparameters args that we have used are in Table 8. Then we began our experiments by translating the dataset using both an Italian and an English prompt (See tab. 7). Subsequently, we translate the dataset using **deepseek-r1-distill-llama-70b** and **gemma2-9b-it** as LLMs and **nllb-200-distilled-600M** as Transformer, in both its non-finetuned and fine-tuned versions. In particular, we have generated a total of 26 translation files. We then performed the tournament-style selection to identify the optimal number of shots for each LLM. The winners were LLaMA 2 K=1 and Gemma 2 K=3. It is important to note that the tournament selection using Prometheus is non deterministic, in fact we ran a small tournament using only a subset of sentences (for computational issues) in order to see if Prometheus judgements are consistent among different trials. In Table 3, it is possible to notice that we got a different winner in each trial of the tournament. Finally, **Prometheus** evaluates the four best models selected from the previous steps using the described metrics. Additionally, we manually assessed the first 20 translations produced by each model to analyze human correlation with the automatic scores, using the **Cohen's Kappa Score**.

4 Results

From the results in Table 5 we can notice that the model LLaMA 2 has the best performances across all models. One concern arises from the Prometheus evaluations of the NLLB-200 Transformer, both fine-tuned (SFT) and non fine-tuned. While human evaluations show a clear improvement in translation quality after fine-tuning with a 11.85% improvement, Prometheus gives them scores that are very close with only a 5.43% of improvement. Such behavior lowers the reliability of Prometheus as a standalone Judge of translation quality, especially when evaluating subtle improvements resulting from model fine-tuning. The Gemma 2 model presents a big mismatch between human and Prometheus evaluations: between Gemma 2 that has 9B parameters we will expect a more advantage with respect the Transformer that has only 600M parameters. In fact, according to our human evaluation, Gemma 2 performs strongly across most metrics, indicating that human evaluators perceive its translations as fluent and semantically complete. In Fig. 5 is possible to notice that the SFT Transformer reaches almost the same mean score of Gemma 2 LLM in two main categories: Modern Structural Effectiveness and Lexical Modernization. This because the two are the main features that the Transformer have learned from our dataset. In Fig. 6, instead, is possible to see that the mean scores are very different with respect to the previous histograms, another reason to state that Prometheus votes are very different from human votes. The same behavior can be noticed in Fig. 7. For instance in the sentence 9 it is very unlikely that a sentence with low votes on meaning preservation, grammar, completeness and lexical modernization can get the maximum vote on structural effectiveness (i.e. the human votes are more reliable across categories with respect to Prometheus that is more "random"). Another reason to that is visible in Table 6, in which the Cohen's Kappa Scores are very low.

Author	Title	Old	Modern
Ludovico Ariosto	*Orlando Furioso*	"Dirò d'Orlando in un medesmo tratto cosa non detta in prosa mai, né in rima..."	"Al tempo stesso racconterò di Orlando una cosa che non è mai stata detta..."
Dante Alighieri	*Divina Commedia*	"salimmo sù, el primo e io secondo, tanto ch'i' vidi de le cose belle..."	"salimmo in alto, lui per primo e io per secondo, finché vidi le cose belle..."
Francesco Petrarca	*Canzoniere*	"Erano i capei d'oro a l'aura sparsi che 'n mille dolci nodi gli avolgea..."	"I capelli biondi erano sparsi al vento che li avvolgeva in mille dolci nodi..."
Giovanni Boccaccio	*Decameron*	"Dico adunque che già erano gli anni della fruttifera incarnazione..."	"Dico dunque che erano già trascorsi milletrecentoquarantotto anni..."

Table 1: Our created dataset: **`oldIT2modIT`** dataset

Author	Date	Region	Sentence	Modern
Brunetto Latini	1260–61	fior.	"quella guerra ben fatta l'opera perché etc. Et dall'altra parte Aiaces era uno cavaliere franco..."	"quella guerra fu ben condotta per via delle imprese compiute, etc. Dall'altra parte, Aiace era un cavaliere coraggioso..."
Bono Giamboni	1292	fior.	"crudele, e di tutte le colpe pigli vendetta, come dice la legge..."	"crudele, e si vendica di tutte le colpe, come prescrive la legge..."
Valerio Massimo (red. V1)	1336	fior.	"Non d'altra forza d'animo fue ornato Ponzio Aufidiano..."	"Non di minor forza d'animo fu dotato Ponzio Aufidiano..."
Lucano volg. (ed. Marinoni)	1330/40	prat.	"Se questo piace a tutti e se 'l tempo hae bisogno d'avere Pompeio per cavaliere..."	"Se questo piace a tutti e se il tempo ha bisogno di avere Pompeo come cavaliere..."

Table 2: Original dataset with our gold translations: **`dataset_gold.csv`** dataset

Tournament trial	Winner LLaMA	Winner Gemma
1	*LLaMA2 K = 3*	*Gemma K = 2*
2	*LLaMA2 K = 1*	*Gemma K = 5*
3	*LLaMA2 K = 0*	*Gemma K = 1*

Table 3: Example of stochasticity of prometheus evals among different trials of tournament selection, here we performed it for only 3 sentences

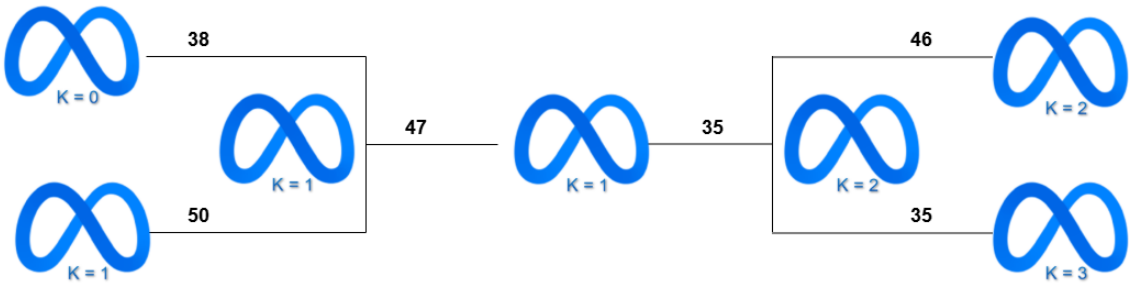

Figure 1: **LLaMA2 tournament results:** in the figure you can see the score obtained by the different configurations from zero-shot (K = 0) to three-shot. The first stage of the tournament was between zero-shot vs one-shot and two-shot vs three-shot. The next stage was between one-shot vs two-shot. The final winner is one-shot. In the figure you can see that the scores do not add to 97, this is due to the fact that prometheus sometimes do not answer in the appropriate manner and thus no score is assigned.

Criterion	Scoring Description (1–5)
Meaning Preservation	1: Completely diverges in meaning; major misinterpretations or hallucinations. 2: Core ideas are misrepresented or lost; significant meaning shift. 3: Mostly correct meaning, but with noticeable errors (minor shifts or omissions). 4: Meaning is preserved with minor discrepancies or nuances lost. 5: Fully faithful to the original meaning.
Grammar	1: Barely readable or nonsensical. 2: Grammatically flawed throughout; affects understanding. 3: Minor grammatical or syntactic errors, but understandable. 4: Fluent with native-like grammar and syntax; minor oddities. 5: Perfectly fluent and natural in the target language.
Modern Structural Effectiveness	1: Structure is archaic or rigid; hard to follow. 2: Structure is awkward or overly literal; unnatural for modern Italian. 3: Mixed structure; understandable but includes dated or clumsy phrasing. 4: Mostly fluent and modern; minor awkwardness or stiffness. 5: Fully natural and idiomatic; clear and fluent for contemporary Italian.
Completeness	1: Large parts missing or major hallucinated content. 2: Multiple omissions or additions that alter meaning. 3: Minor omissions or additions that don't significantly alter meaning. 4: Nearly complete with only slight trimming. 5: Fully complete with no omissions or additions.
Lexical Modernization	1: No modernization; reads like the original. 2: Mostly archaic or unnatural word choices. 3: Mixture of modern and archaic terms. 4: Slightly dated or formal but acceptable in modern Italian. 5: Modern terms feel natural, idiomatic, and current.

Table 4: Rubric scoring criteria used for model evaluation. Each category is scored from 1 (poor) to 5 (excellent).

Model	Judge	MP	GR	MSE	CO	LM	TOT
LLaMA 2 70b (distilled) Few-Shot K=1	Our	80	81	79	83	64	387
	Prometheus	*67*	*70*	*60*	*70*	*59*	*326*
GEMMA 2 9b Few-Shot K=3	Our	79	83	67	87	55	371
	Prometheus	*61*	*48*	*60*	*48*	*62*	*279*
NLLB-200 Transformer 600M (distilled)	Our	46	60	61	57	46	270
	Prometheus	*53*	*48*	*62*	*53*	*60*	*276*
NLLB-200 Transformer 600M (distilled) SFT	Our	53	65	67	61	56	302
	Prometheus	*59*	*51*	*58*	*58*	*65*	*291*

Table 5: Sum of human (Our) and Prometheus evaluation scores for each model and metric. MP = Meaning Preservation, GR = Grammar, MSE = Modern Structural Effectiveness, CO = Completeness, LM = Lexical Modernization.

Model	MP	GR	MSE	CO	LM
DeepSeek	0.118	-0.027	-0.030	-0.063	-0.073
Gemma	0.248	-0.005	0.146	-0.041	-0.014
Transformer (FT)	0.014	-0.114	-0.032	0.063	0.203
Transformer (Non-FT)	0.065	0.130	0.012	0.133	0.190

Table 6: Cohen's Kappa scores between Prometheus and human evaluations for each model and rubric. MP = Meaning Preservation, GR = Grammar, MSE = Modern Structural Effectiveness, CO = Completeness, LM = Lexical Modernization.

Key	Content
`system_template_it`	Sei un esperto linguista italiano.
`system_template_en`	You are an expert Italian linguist.
`user_template_it`	Traduci da italiano antico a italiano moderno, tenendo conto dei modi di dire e scrivere dei tempi medievali e rinascimentali, scrivimi solo la traduzione senza aggiungere altro. `{examples}` Antico: '`{old_sentence}`' Moderno:
`user_template_en`	Translate from Old Italian to Modern Italian, taking into account the ways of speaking and writing of the medieval and Renaissance periods, just write me the translation without adding anything else. `{examples}` Archaic: '`{old_sentence}`' Modern:

Table 7: Templates for Italian Linguist Prompting

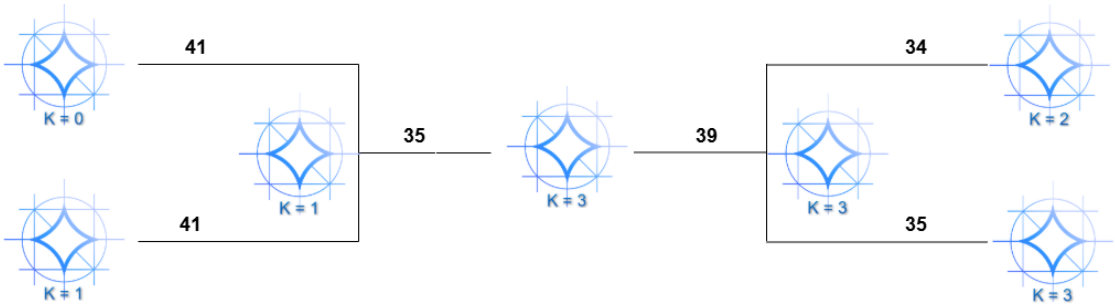

Figure 2: **Gemma2 tournament results:** in the figure you can see the score obtained by the different configurations from zero-shot (K = 0) to three-shot. The first stage of the tournament was between zero-shot vs one-shot and two-shot vs three-shot. The next stage was between one-shot vs three-shot. The final winner is three-shot. In the figure you can see that the scores do not add to 97, this is due to the fact that prometheus sometimes do not answer in the appropriate manner and thus no score is assigned.

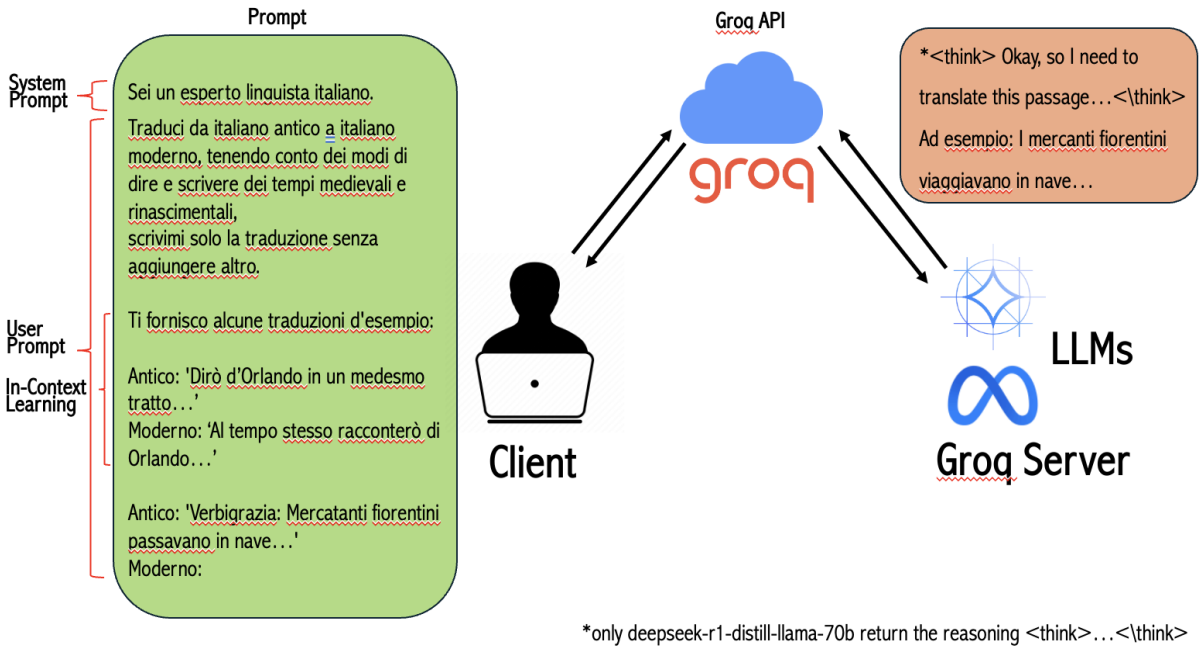

Figure 3: LLMs translation pipeline using Groq API

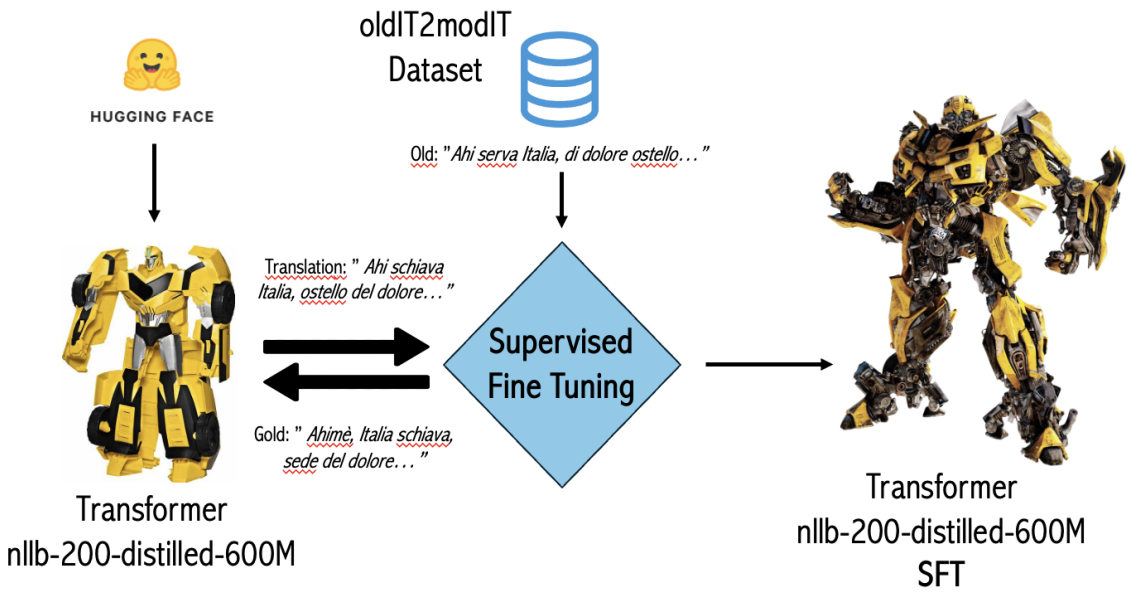

Figure 4: Transformer Supervised Fine Tuning mechanism

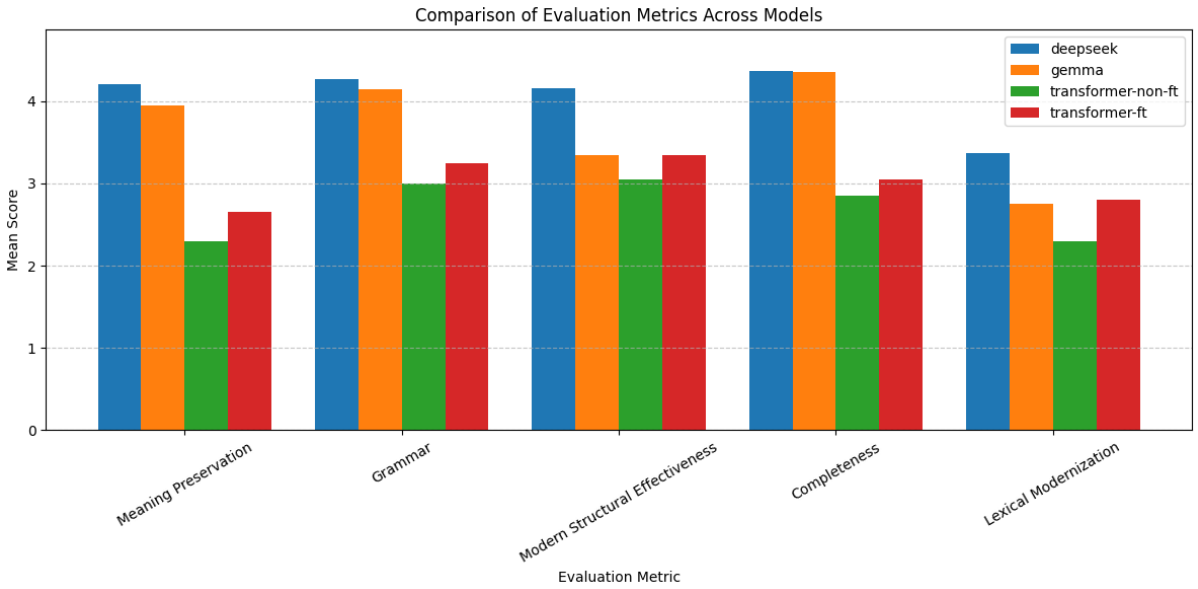

Figure 5: Evaluation on the first 20 translations made by us for the four models

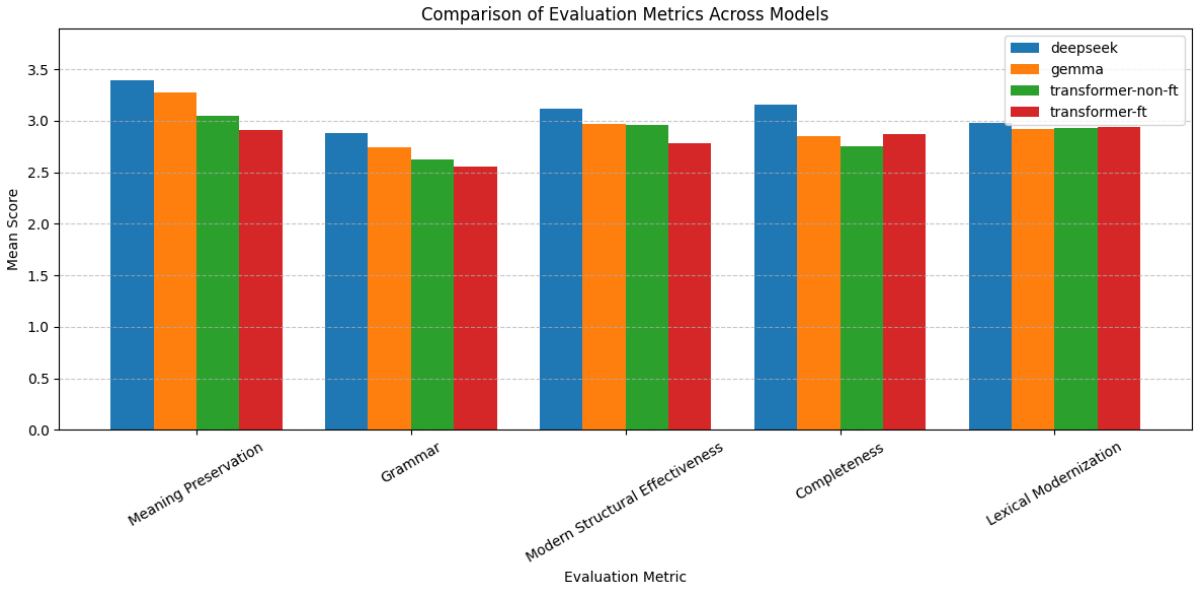

Figure 6: Evaluation on the whole translations made by prometheus

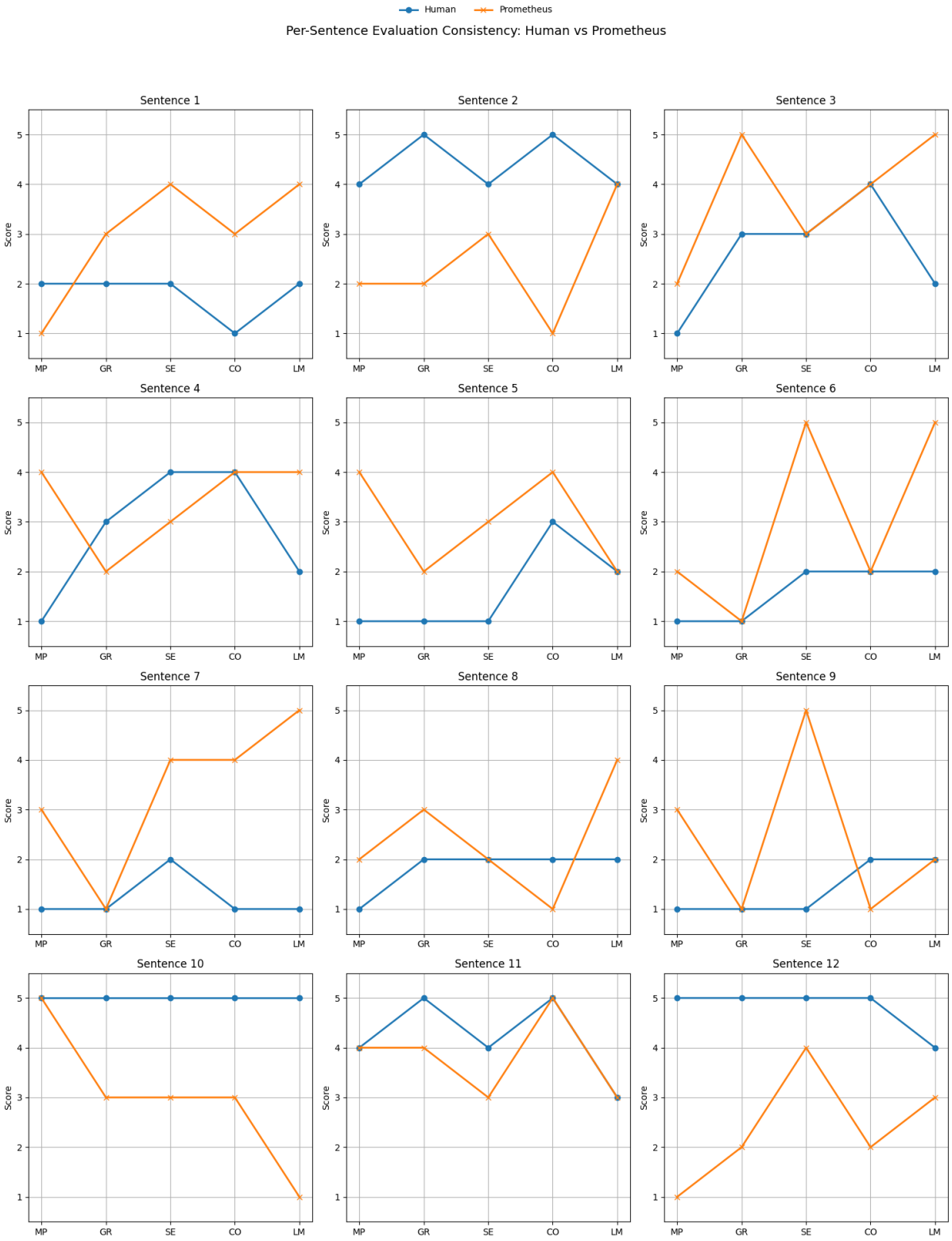

Figure 7: **Evaluation consistency between prometheus and human for 12 sentences on the transformer non fine tuned dataset:** you can see that prometheus votes are more randomic across categories.

Fine-Tuning Arg	Value	Explanation
per_device_train_batch_size	5	Number of samples processed in each training step per GPU. With 200 examples, this leads to 40 steps per epoch.
num_train_epochs	3	Number of full passes through the dataset. Total training steps = $3 \times 40 = 120$.
logging_steps	10	Frequency (in steps) at which logs (e.g., loss) are printed.
save_steps	500	Frequency of checkpoint saving. Since training ends at step 120, only the final checkpoint is saved.
save_total_limit	1	Limits the number of saved checkpoints. Older ones are deleted, keeping only the latest.

Table 8: Fine-tuning hyperparameters