Metaphors Unveiled: Exploring Language Models for Figurative Text and The Gap Between Sentence Probabilities and Decoded Text

David Guzman Piedrahita
22-737-571
david.guzmanpiedrahita@uzh.ch

Rajiv Bains
22-722-870
rajiv.bains@uzh.ch

Lucas Krauter
22-740-369
lucas.krauter@uzh.ch

Abstract

Figurative language, including metaphors and similes, enriches human communication by adding depth and nuance. While the recent surge in Natural Language Processing (NLP) has advanced the field, particularly in the realm of literal language interpretation, there remains a research gap in the comprehension of figurative language by large language models (LLMs). This study delves into the proficiency of LLMs, including FLAN-T5 large, Mistral, and Phi1.5B, in processing and simplifying sentences with metaphors. We replicated the results from (Liu et al., 2022) as our benchmark. Subsequent experiments assessed the effectiveness of seq2seq fine-tuning, prompt engineering, and a combination of both with FLAN-T5 large. Notably, prompt-engineering alone augmented model accuracy from 61.5% to 71.4%. Furthermore, human evaluations depicted a stark contrast between the metric proposed by (Liu et al., 2022) and the human-perceived correctness of metaphor interoperation, where prompt-engineering plummets to 8% whereas a sequence-to-sequence fine-tuned Flan-T5-Large achieved 64%. The implications of these findings underscore the complexities of figurative language processing by LLMs and the potential pitfalls of relying solely on automated metrics as well as the gap between sentence probabilities and decoded outputs. Code is available here.

1 Introduction

Figurative Language exerts influence in human language and conversation. The use of similes, metaphors and so forth sends an additional message to the receiver that aids to consolidate their understanding. The recent growth of Natural Language Processing (NLP) has been focused in the area of translation and interpretation, namely with literal (non-figurative) language. However, there is a lack of research and thus understanding of how large language models (LLMs) process figurative text.

This paper explores the performance of a pre-existing large language models —such as FLAN-T5 large (Chung et al., 2022), Mistral (Jiang et al., 2023) or Phi1.5B (Li et al., 2023)— to process sentences containing metaphors and generate a simplified version of the sentence without figurative language.

First, we reproduce the results from (Liu et al., 2022) and use them as a baseline comparison against our model. Then, we held experiments which evaluated the performance of FLAN-T5 large in a baseline scenario, with seq2seq finetuning and with prompt engineering.

The behavior of the models reveals that using a prompt alone increases the likelihood the model assigns to the sentence containing the concatenation of the metaphor and its correct interpretation, and decreases the probability of the concatenation with the incorrect one. Other experiments show that fine tuning the model alone leads to more limited gains in metaphor understanding and may even degrade performance in some cases.

Our highest score, achieved with Flan-T5-Large (780M parameters) via manual prompt-engineering, is 71.4%, which outperforms all non-fine-tuned models in (Liu et al., 2022) and is only a few points away from GPT-3 Babbage fine-tuned (1.3B parameters) which scores 73.97%.

Furthermore, our project aimed to delve into how various models comprehend metaphors across diverse settings. Initially, we employed a metric, proposed by (Liu et al., 2022), which measures accuracy by assessing whether a model assigns a higher likelihood to the correct versus the incorrect metaphor interpretation in the dataset. To further our investigation, we also incorporated human evaluations to gauge the accuracy of models when tasked with generating metaphor explana-

tions, rather than just assigning likelihoods.

Our findings revealed that our prompt-engineering led to a substantial increase in model accuracy (61.5% to 71.4%) but when the generated answers were evaluated by humans, accuracy had dropped to 8%. Contrarily, our Flan-T5-Large model, fine-tuned with a sequence-to-sequence objective, showcased a far superior performance of 64%. This is noteworthy, especially considering its modest 2-point improvement (60.8% to 62.8%) as per the metric employed in (Liu et al., 2022).

2 Datasets

We used the "Figurative QA" dataset which includes metaphors in text format along with their targets (i.e., their correct interpretation in natural language). The dataset is composed of 10,256 human-written metaphors that are paired as a Winograd schema (see table 2). It is available to download from huggingface (https://huggingface.co/datasets/nightingal3/fig-qa).

3 Pre-processing

3.1 Data Pre-processing

The pre-processing stage was divided into two separate sections: the first being data pre-processing. This involved the tokenization of data where we employed the tokenizer the respective model. A sequence length maximum of 40 characters was chosen since it ensures that the metaphor explanations are neither truncated nor overly lengthy, which otherwise might hamper the model's quick convergence during fine-tuning.

Initially, the data handling involved reproducing the strategy from the original paper, where the metaphor was concatenated with the correct interpretation. This configuration enabled the fine-tuning of autoregressive models on a designated causal language model training objective. However, given the constraints with limited training capabilities and GPU compute, this approach was substituted by a sequence-to-sequence training strategy. Here, the metaphor served as the input, while the correct interpretation was set as the target. This mirrors the machine translation paradigm; the difference being the conversion from figurative language to plain English as opposed to translation between languages.

Two prompt formats were incorporated: a start prompt and a middle prompt. The former entails a string being prepended to the input and generally provides additional instructions to the model. This was especially relevant for T5 which was pretrained with prefixed directives. The middle prompt is situated between the metaphor and its interpretation and thus offers contextual clarity. Examples of the middle prompt include the phrases; "this is to say" or "this means that."

3.2 Model Pre-processing

In the model preprocessing stage, for the largest models we didn't directly employ the model as provided by Hugging Face. To enhance computational efficiency and memory usage, especially for larger models, we adopted a four-bit quantization approach, which accelerates data propagation through the model compared to the conventional 16 or 32-bit schemes. Specifically, for these models, we leveraged Qlora fine-tuning (Dettmers et al., 2023), a technique allowing for the selective fine-tuning of a low-rank adapter that alters the model's attention mechanism.

For the components subject to optimization (namely, the adapter), we opted for a 16-bit representation over the typical 32-bit to economize memory in light of our computational limitations.

While these methodologies might pose potential trade-offs in performance, they facilitated the deployment of a more condensed model version, making it feasible to harness larger models.

4 Model Training

4.1 Reproduction of models and results from earlier work, now with adapters

In our preliminary efforts, the methodology was centered around replicating the fine-tuning strategy of the foundational paper (Liu et al., 2022). The data strategy used involved concatenating the metaphor with its correct interpretation, training autoregressive models on a causal language modeling objective. Our innovation involved the integration of Qlora adapters to make the training more memory efficient with fewer parameters to adjust. Initially, the models Mistral, Synthia, and TinyLlama were taken into consideration (see table 3).

Before fine-tuning, the Mistral model exhibited an accuracy of 60.8%. Following the application of adapters to its attention mechanism over 500 batches of ten samples at a learning rate of 0.001, there was no improvement in accuracy after 500

batches, remaining at 60.8%. Synthia, a derivative of Mistral, was trained for 40 batches and demonstrated no change in accuracy, remaining static at 63.1%. TinyLlama, a smaller variant of Llama1, possessing 1.1 billion parameters and optimized for chat, showed no change in accuracy after 70 batches of 20 samples each, maintaining at 56%.

These consistent results could suggest several hypotheses. More epochs or batches might be required to initiate training. Furthermore, greater priority might be needed for the adapter as it integrates with the existing attention mechanism via the Qlora hyperparameters, which represents a direction for future work (see 6).

Subsequently, we recreated from scratch the traditional fine-tuning setting from paper (Liu et al., 2022), to determine whether training the entire set of parameters would lead to different results, with the added bonus of Qlora fine-tuning that only targets the attention mechanism. Due to computational limitations, however, the models mentioned above could not be considered; in their place the smaller models Phi 1.5B and Flan-T5_Base.

The Phi model, encompassing 1.5 billion parameters, was fine-tuned over three epochs with a batch size of 30 samples and a learning rate of 0.001. The accuracy observed a minute increase from 60.6% to 60.8%. Fine-tuning for 1.64 epochs with a batch size of eight, this model's accuracy dropped from 56.6% to 46.6% (see table 4).

Given the mismatch in results, a further three models were assessed using the original code provided by the authors.

GPT Neo SSM Model was fine-tuned over seven epochs with a batch size of four samples, early stopping patience of four batches, and a learning rate of 10^{-5}, the model's accuracy saw a marginal increase from 52.8% to 53.6%. T5 Small Model using the same hyperparameters as GPT Neo but trained over two epochs declined from 52.2% to 48.6% (see table 5).

These explorations lead to several key conclusions:

1. Setting up the training requires meticulous attention, and without strategic hyperparameter tuning, optimal results are elusive. However, due to computational constraints and the project's timeframe, a comprehensive hyperparameter search couldn't be conducted.

2. Excluding Qlora fine-tuning, the models traditionally fine-tuned were substantially smaller than those in the original paper. This reduced parameter count may result in diminished model expressivity, explaining the limited performance increase of GPT Neo SSM, despite it belonging to a family of models covered in (Gao et al., 2020).

4.2 Sequence to Sequence Fine-tuning for Flan-T5-Large

In this phase of our study, we undertook fine-tuning as a strategy to improve baseline performance. As explained in 3.2, we tokenized the data and adapted it for a sequence-to-sequence objective. The model was preprocessed to employ quantization, and fine-tuning was constrained to the adapters associated with the values and queries, excluding the keys of the attention mechanism(Dettmers et al., 2023).

- **Hardware and Model**: We utilized two T4 GPUs, each equipped with 16GB memory. The model of choice was T5-large (FLAN-T5 large), which was then fine-tuned to our seq2seq objective.

- **Initial Hyperparameters**:
  - Learning Rate: 0.001.
  - Batch Size: 10 samples.
  - Early Stopping: set to three, evaluated every 20 batches.

With these settings, convergence was achieved after 500 batches, each with 10 samples, which is notably less than a full epoch. The resulting performance improvement was an increase in accuracy from 60.8% to 62.8%, upon validation with 500 samples (see table 6). We explored different combinations of hyperparameters to optimize our results (see table 7):

1. Extended Fine-tuning: Early stopping patience was expanded to seven and convergence spanned three epochs; a dramatic difference to the 500 steps in the initial experiment. However, this resulted in an accuracy decrease to 54%.

2. Batch Size Augmentation: Given our memory constraints, the maximum feasible batch size was 150 samples. A significant batch size increment would potentially lead to more stable gradient descent steps, due to exposure to more data in each step. The model's accuracy degraded to 58.8% after eight epochs. Interestingly, this degradation was less pronounced than the one observed over three epochs which could imply that a larger batch size

might positively influence accuracy performance after convergence.

3. Model Size: We applied the initial hyperparameters on T5-base but this led to a drop in accuracy from 55% to 54.6%. This underscores the potential influence of model expressivity on fine-tuning efficacy. The highest accuracy was achieved before the completion of a single epoch which suggests that a single pass through the data might suffice for the model to maximize improvement under a given set of hyperparameters; this is a finding consistent with other experiments using QLoRA (Dettmers et al., 2023).

We found that the best performing learning rate is 0.001. Rates exceeding 0.001 rendered the training process too rapid and unstable for effective convergence while anything below caused sluggish training without performance enhancement.

4.3 Prompt Engineering

4.3.1 Background and Motivation

While our fine-tuning experiments produced some enhancements in model performance, an alternative approach of prompt engineering we pursued stemmed from hints in (Liu et al., 2022). Although the original paper indicated only a modest 1-2% improvement through this method, we embarked on a more extensive exploration of this area.

4.3.2 N-shot Setting Experiment

Initially, instead of traditional prompts, we employed an N-shot setting. Here, during inference, the model was presented with samples encompassing a metaphor, its interpretation, and a concise explanation of said interpretation. Table 1 elucidates that, upon escalating the sample count from one to twenty, accuracy diminishes. T5 base started with a 55% accuracy rate, deteriorating by ten percentage points to 45% at twenty samples (see table 8). This underscores that T5 base may not efficiently harness an N-shot setting for this particular test.

4.3.3 Prompt Engineering Experiments

Pivoting from the N-shot approach, we transitioned to actual prompt engineering, crafting prompts that enhance the models' contextual understanding of the task at hand. This strategy was employed for the T5 base, as well as some larger counterparts —Mistral chat and T5 Large—, albeit the latter were quantized to a four-bit precision.

Our results revealed that the original paper's prompt, "this is to say," was quite effective. In instances, it caused a six-point enhancement, especially with models like T5 Large Quantized (see table 9). These prompts were concatenated after the metaphor, serving as a linguistic bridge to fluently transition from metaphor to explanation, as suggested by (Liu et al., 2022).

Additionally, we innovated with templates comprising prefixes to the metaphor itself, in addition to prompts concatenated right after it. Such a design is pivotal since the T5 model family inherently attends to prefixes ahead of specific inputs, as they reflect task-related directives used in its training (Raffel et al., 2020). This revamped approach ushered in a spike from a 61.5% baseline accuracy to a maximum of 71.4% – credited to the two top-performing template prompts (See table 10).

However, we must note that due to computational limitations, not all prompts were tested across identical sample sizes. The accompanying tables delineate the sample counts for each prompt, with subsets randomly selected from the validation set.

4.3.4 Discussion and Insights

Prompt engineering's superior performance, compared with fine-tuning, corroborates previous research assertions on large language models (Radford et al., 2019). Often, prompting emerges as the first-line strategy when tailoring a large language model for specialized NLP tasks. It can yield substantial performance gains without incurring the computational overhead linked to fine-tuning. While Qlora attempts to mitigate the computational toll of fine-tuning, prompting remains the initial go-to technique for model customization.

Additionally, our findings underscore the model's susceptibility to noise. A mere data-cleaning activity, such as standardizing the presence or absence of a period post-metaphor, can foster performance improvement (refer to section 5.1). This not only highlights the model's sensitivity to specific prompts but also underscores that optimal prompts might not always be self-evident.

4.4 The gap between sentence probability and decoding: human evaluation

we subjected our sequence-to-sequence fine-tuned model, the baseline (not fine-tuned), and the optimally prompted model to human evaluations, aiming to understand the quality and correctness of

	Metric	Human Score
Baseline	68.8%	1%
Best prompt	74.4%	8%
Fine Tuned	62.8%	64%

Table 1: Comparison of Evaluation Methods: the gap between sentence probabilty and decoded outputs

their interpretations. An author corrected 50 predictions from the validation set for each variant. While the original metric favored the prompted model, human evaluations ranked the fine-tuned version highest at 64% accuracy. In stark contrast, the baseline scored only 2%, and the prompted variant managed 8% (see table 1).

5 Further Discussion

In this chapter, we will analyse the implications of our findings.

5.1 Data pre-processing approach

As a consequence of our research process, it was observed that sometimes the sentences and metaphors were incorrectly segregated by periods or other punctuation marks. A few metaphors ended with a period in the dataset, whilst others did not. Consequently, the current benchmark software sometimes results in two points separating the metaphor from the interpretation. As the crowd workers created the dataset, its accuracy may vary across different areas owing to discrepancies in the labeling process.

After removing punctuation and concatenating the sentences correctly with a single full stop, we found that the accuracy increased to 63%, compared to the base accuracy of 60.80% for Flan-T5-Larged quantized to 4-bit precision. This can be seen in table 10 using the pattern "{context}. {sentence}". We think that writing sentences properly, without excessive special characters, can enhance the model's comprehension of the metaphor structure and improve accuracy.

5.2 Certain types of metaphors

When validating the results, we observed that in some situations, the model interpreted the metaphor incorrectly but still generated a valid interpretation from another perspective. The model was not able in all scenarios to interpret a metaphor which typically was longer in length or included a color/movement/ 'an a/an' or an emotion. This is because the model requires a deeper understanding of the semantics, (Liu et al., 2022).

6 Future Work

Our experiment shows that sequence-to-sequence training markedly improves metaphor interpretation generation, but it does not enhance the metric (which is solely focused on sentence probabilities) introduced in (Liu et al., 2022). We hypothesize that distinct decoding techniques utilized by various models might significantly influence this gap and would like to explore this avenue.

Additionally, an area of exploration involves fine-tuning the varied hyperparameters provided by the QLoRA framework. This includes determining which aspects of the attention mechanism (keys, queries, values, and linear transformations) to fine-tune, (Dettmers et al., 2023). Experimenting with diverse schedulers and batch sizes may also offer insights.

Another prospective enhancement lies in altering the training objective. Given that the dataset includes incorrect interpretations, we propose to formulate a contrastive learning task for FLAN-T5 and related models and integrate a pseudo-contrastive objective for autoregressive models. This may bolster performance by differentiating correct from incorrect instances. This approach has already demonstrated efficacy for masked models (Liu et al., 2022).

Prompt engineering could be further extended into prompt tuning. Unlike traditional prompt engineering, prompt tuning replaces the natural language sentence prompt with an embedding learned through optimization. This approach neither alters the model's weights akin to fine-tuning nor uses a natural language prompt as in prompt engineering. Nevertheless, it has demonstrated enhanced performance across various NLP domains.

Based on this, we would be keen to delve deeper in collaboration with the University of Zurich. We aspire to develop this research further, positioning it as a candidate for academic publication.

References

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, Albert Webson, Shixiang Shane Gu, Zhuyun Dai, Mirac Suzgun, Xinyun Chen, Aakanksha Chowdhery, Alex Castro-Ros, Marie Pellat, Kevin Robinson, Dasha Valter, Sharan Narang, Gaurav Mishra, Adams Yu, Vincent Zhao, Yanping Huang, Andrew Dai, Hongkun Yu, Slav Petrov, Ed H. Chi, Jeff Dean, Jacob Devlin, Adam Roberts, Denny Zhou, Quoc V. Le, and Jason Wei. 2022. Scaling instruction-finetuned language models.

Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. 2023. Qlora: Efficient finetuning of quantized llms.

Leo Gao, Stella Biderman, Sid Black, Laurence Golding, Travis Hoppe, Charles Foster, Jason Phang, Horace He, Anish Thite, Noa Nabeshima, et al. 2020. The pile: An 800gb dataset of diverse text for language modeling. *arXiv preprint arXiv:2101.00027*.

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.

Yuanzhi Li, Sébastien Bubeck, Ronen Eldan, Allie Del Giorno, Suriya Gunasekar, and Yin Tat Lee. 2023. Textbooks are all you need ii: phi-1.5 technical report.

Emmy Liu, Chen Cui, Kenneth Zheng, and Graham Neubig. 2022. Testing the ability of language models to interpret figurative language.

Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer.

A Appendix

All of the tables in the appendix are merely to support the reader. The content they convey is present, in writing, in the 5 pages of the report. Should it be necessary, this part can be safely ignored.

Start Phrase	Ending 1	Ending 2	Label
Her word had the strength of titanium.	Her promises can be believed.	Her promises cannot be trusted.	0
His kisses have the passion of lovers meeting after a long separation.	His kisses are demonstrative and intense.	His kiss is unemotional.	0

Table 2: Random samples of the training set

Model	Initial Accuracy (%)	Batches	Final Accuracy (%)
Mistral	60.8	500 (10 samples)	60.8
Synthia	63.1	40	63.1
TinyLlama	56.0	70 (20 samples)	56.0

Table 3: Results after applying Qlora adapters to models' attention mechanisms

Model	Initial Accuracy (%)	Epochs (Batch size)	Learning Rate	Final Accuracy (%)
Phi 1.5B	60.6	3 (30 samples)	0.001	60.8
Flan-T5_Base	56.6	1.64 (8 samples)	0.001	46.6

Table 4: Results after recreating traditional fine-tuning settings with Qlora fine-tuning

Model	Initial Accuracy (%)	Epochs (Batch size)	Final Accuracy (%)
GPT Neo SSM	52.8	7 (4 samples)	53.6
T5 Small	52.2	2 (4 samples)	48.6

Table 5: Results of models fine-tuned using the original code from authors

Specification	Value
Hardware	2x T4 GPUs with 16GB
Model	T5-large (FLAN-T5 large)
Initial Learning Rate	0.001
Batch Size	10 samples
Early Stopping	3 (evaluated every 20 batches)
Convergence	500 batches (less than a full epoch)
Initial Accuracy	60.8%
Final Accuracy	62.8%

Table 6: Details of the baseline setup and achieved results.

Experiment	Specification	Result
Extended Fine-tuning	Early stopping: 7	54% accuracy after 3 epochs
Batch Size Augmentation	Batch Size: 150	58.8% accuracy after 8 epochs
Model Size	Model: T5-base	54.6% accuracy

Table 7: Results from the series of hyperparameter optimization experiments.

Example Count (n-shot)	Accuracy
0	0.55
1	0.54
2	0.45
5	0.47
10	0.46
20	0.45

Table 8: Accuracy vs. Example Count (n-shot) with T5-base

Model	Prompt	Samples	Accuracy
T5-base	/	100	0.55
T5-base	”Whereas”, ”Moreover”, ”Furthermore”	100	0.59
T5-base	/	1000	0.546
T5-base	”That is to put it mildly”	1000	0.558
T5-large / quantized	/	400	0.615
T5-large / quantized	”This is to say”	400	0.675
Mistral-Chat-Tuned / quantized	”This is to say”	400	0.7075
Mistral-Chat-Tuned / quantized	/	500	0.672

Table 9: Model Accuracy Results on PROMPTS

Pattern	Samples	Accuracy
“{context}\nCan we infer the following?\nThis is to say\n{sentence}.“	70	0.714
“Can you interpret the following: {context}. What is the answer??\n{sentence}“	70	0.714
“{context}. {sentence}.“	70	0.629
“{context}\nThis is to say:\n{sentence}“	70	0.686
“{context}\nCan we draw the following conclusion?\nThis is to say \n{sentence}“	500	0.698
“{context}. {sentence}.“	500	0.63
“{context}. That is to say, {sentence}.“	500	0.618
“{context} that is to say {sentence}.“	500	0.572
“{context}\nCan we infer the following?\nThis is to say\n{sentence}.“	500	0.658
“Can you interpret the following: {context}. What is the answer??\n{sentence}“	500	0.622

Table 10: Template Accuracy Results with T5-large Quantized