It is not a piece of cake for GPT: Explaining Textual Entailment Recognition in the presence of Figurative Language

It is not a piece of cake for GPT: Explaining Textual Entailment Recognition in the presence of Figurative Language

Giuseppe Gallipoli and **Luca Cagliero**
Politecnico di Torino
{giuseppe.gallipoli, luca.cagliero}@polito.it

Abstract

Textual Entailment Recognition (TER) aims to predict whether a pair of premise-hypothesis sentences represents an entailment, a contradiction, or none of the above. Addressing TER in the presence of figurative language is particularly challenging because words are used in a way that deviates from the conventional order and meaning. In this work, we investigate the capabilities of Large Language Models (LLMs) to address TER and generate textual explanations of TER predictions. First, we evaluate LLM performance in Zero- and Few-Shot Learning settings, with and without using Chain-of-Thought prompting. After identifying the best prompts, we highlight the settings in which in-context learning is beneficial. The closed-source models GPT-3.5 Turbo and GPT-4o show unexpected limitations compared to significantly smaller open-source LLMs. Next, we thoroughly analyze the effect of LLM Fine-Tuning, showing substantial improvements in the quality of TER explanations compared to Zero- and Few-Shot Learning. Notably, 9 billion parameter open-source LLMs demonstrate again competitive performance against larger closed-source models. Finally, we compare our LLM-based approach with the state-of-the-art DREAM-FLUTE and Cross-Task architectures. The results show significant performance improvements, particularly in the quality of the generated explanations.

1 Introduction

Figurative language is quite common in both written and spoken conversations. In natural language texts, examples of figurative language such as metaphors, similes, and sarcastic expressions are commonly used to convey opinions, ideas, or emotions (Holme, 2004). Due to the high variety and inherent complexity of these non-literal expressions, state-of-the-art Natural Language Understanding (NLU) and Inference models, such as Transformers (Vaswani et al., 2017), often struggle with texts and dialogues including figurative language (Jhamtani et al., 2021).

Textual Entailment Recognition (TER) is an established NLU task. Given a pair of premise and hypothesis sentences, the goal is to classify the pair as *entailment*, if the hypothesis can be inferred from the premise, *contradiction*, if the premise contradicts the hypothesis, or *neutral* otherwise.

TER on sentences including figurative language is known to be particularly challenging (Chakrabarty et al., 2021) because it not only requires advanced predictive capabilities but also an advanced comprehension of the underlying message. Language Models (LMs) often struggle in understanding unconventional expressions that are crucial for TER prediction.

TER and Explanation (TER+E) is an extension of the classical TER task where the LM is asked to provide end users with a textual explanation of the TER prediction (Chakrabarty et al., 2022). TER+E combines the challenges of the NLU task with a generative step. State-of-the-art approaches, i.e., Cross-Task (Bigoulaeva et al., 2022) and DREAM-FLUTE (Gu et al., 2022b), leverage LMs fine-tuned on benchmark data, e.g., the FLUTE dataset released in 2022. More recently, the diffusion of Large Language Models (LLMs) has opened the way for a new paradigm of joint language understanding and generation. Although several large-scale pre-trained models have been released, to the best of our knowledge the capabilities of LLMs on TER+E are still unexplored.

This paper presents LLM-based approaches to address TER and Explanation. Firstly, it explores the zero-shot performance of several LLMs, both open-source and proprietary. The aim is to compare the models' robustness to figurative language understanding. The results are contrasting: LLMs generally perform well on TER, but struggle in generating high-quality TER explanations. Notably, open-source 9B parameter LLMs surpass the

Proceedings of the 31st International Conference on Computational Linguistics, pages 9656–9674
January 19–24, 2025.

closed-source models GPT-3.5 Turbo and GPT-4o, demonstrating a superior ability to explain figurative language compared to significantly larger models. Similar results hold in a Few-Shot Learning setting, where we also explore the use of Chain-of-Thought (CoT) prompting. In-context learning has shown to be beneficial to improve zero-shot performance but, unlike state-of-the-art models (e.g., DREAM-FLUTE (Gu et al., 2022b)), in general LLMs do not benefit from CoT because of their robust pre-training. Finally, we present LLM Fine-Tuning for TER+E and test its performance on an open-source benchmark. The results are consistently superior to state-of-the-art TER+E approaches. Again, open-source fine-tuned LLMs turn out to be competitive against proprietary GPT models. The challenges of GPT models are confirmed by their poor performance when used *as an expert reviewer*, i.e., when we ask GPT-4o to revise and potentially correct the outputs generating by smaller, open-source LLMs. GPT tends to unnecessarily correct the TER explanations, likely due to inherent limitations in figurative language understanding.

In summary, this work is the first attempt to use LLMs to address TER+E. Our main findings can be summarized as follows.

- TER+E has shown to be way more challenging than TER only in the presence of figurative language, especially for non-fine-tuned LLMs.
- Prior results reported in Gu et al. (2022b), which show the benefits of providing the model with external knowledge, turn out to be no longer valid for LLMs, whose performance does not improve while adopting CoT prompting.
- Fine-tuned LLMs perform significantly better than all state-of-the-art models on the FLUTE benchmark.
- Closed-source LLMs (GPT-4o and GPT-3.5 Turbo) struggle in generating reliable TER explanations. Even in the Few-Shot Learning setting, the performance of GPT-4o is considerably lower than significantly smaller LLMs.

To foster the reproducibility of our results, the models and code used for the implementation of the presented approaches are publicly available, for research purposes only, under the CC BY-NC-SA license, at https://github.com/gallipoligiuseppe/FigurativeTER-E_LLM.

The remainder of the paper is organized as follows. Section 2 formalizes the figurative language understanding and explanation tasks. Sections 3 and 4 review the related literature and present the LLM-based approaches, respectively. Section 5 describes the main experimental results whereas Section 6 draws conclusions and discusses the future research agenda.

2 Problem formulation

Let us consider a human-curated collection of premise-hypothesis sentence pairs, hereinafter denoted by $\langle \mathtt{pr}_j, \mathtt{hp}_j \rangle$. Each pair corresponds to a distinct natural language instance and is classified as either an Entailment or a Contradiction.[1] We will denote by l_j the binary label (taking value E or C) of an arbitrary pair $I_j = \langle \mathtt{pr}_j, \mathtt{hp}_j \rangle$. Natural language instances contain figurative language. Examples of figurative language included in the sentence pairs are simile, metaphor, idiom, and sarcasm.

Each instance I_j is also annotated with a natural language explanation E_j of the contained figurative language. Importantly, both the label and the explanation are defined in terms of the figurative language expression rather than with respect to any other part of the sentence. Hence, identifying and understanding these unconventional expressions in the sentence pairs is crucial for our purposes.

As an example, a metaphor instance and its corresponding label and explanation is reported below.
Premise: *The promise between us was a poisonous snake.*
Hypothesis: *The promise between us was a flower.*
Label: *Contradiction*
Explanation: *A flower is something that is beautiful and represents growth, while a poisonous snake is something that is dangerous and can harm.*

We formulate the following tasks:

Textual Entailment Recognition (TER): Given an arbitrary instance I_j containing figurative language, predict the value of the corresponding label l_j.

Textual Entailment Explanation (Explanation): Given an arbitrary instance I_j and a predicted label l_j, generate the corresponding natural language explanation E_j.

[1] Hereinafter we will disregard the case of neutral sentence pairs, which is not relevant to our purposes.

We aim to address a combination of the above-mentioned tasks:

Textual Entailment Recognition and Explanation (TER+E): Given an arbitrary instance I_j containing figurative language, predict the value of the corresponding label l_j and generate the corresponding natural language explanation E_j.

Note that, instead of addressing TER and Explanation separately, we aim to exploit the synergies and cross-dependencies between them. Furthermore, unlike standard TER which involves only literal language, the presence of figurative language requires a deeper understanding of the underlying meaning conveyed by non-literal expressions.

3 Related Works

Various English benchmarks for TER in the presence of figurative language have been proposed in literature (Agerri, 2008; Agerri et al., 2008; Chakrabarty et al., 2021; Phelps et al., 2024). They consist of human-curated collections of sentence pairs including figurative language examples. The most common issues with existing benchmarks are (1) The relatively limited number of examples; (2) The focus on a specific type of figurative language, e.g., idioms (Phelps et al., 2024); (3) The lack of natural language label explanations. The FLUTE benchmark (Chakrabarty et al., 2022) overcomes the aforesaid issues by proposing a larger benchmark for TER and Explanation (TER+E). It covers four types of figurative language and includes the textual explanations of the TER predictions.

According to the FigLang 2022 Shared Task report (Saakyan et al., 2022), DREAM-FLUTE (Gu et al., 2022b) and Cross-Task (Bigoulaeva et al., 2022) are the best-performing TER+E approaches. Both of them leverage external models or data sources to make LM outcomes more robust. Specifically, DREAM-FLUTE leverages DREAM (Gu et al., 2022a), i.e., a scene elaboration module. The purpose is to build a mental model of the situation presented in the premise-hypothesis pair that is instrumental for both TER and Explanation. The authors propose five different system versions, all of them based on the fine-tuning of a T5 3B model (Raffel et al., 2020). Specifically, Systems 1 and 2 use original data only, System 3 leverages external knowledge extracted from DREAM, System 4 implements a two-step classify-then-explain approach, and System 5 is an ensemble of the above. Cross-Task (Bigoulaeva et al., 2022) aims to transfer information from related tasks to improve TER+E performance. In detail, they use T5 by transferring knowledge from two external datasets, i.e., IMPLI (Idiomatic and Metaphoric Paired Language Inference) (Stowe et al., 2022) and SNLI (Natural Language Inference with Natural Language Explanation) (Camburu et al., 2018).

To the best of our knowledge, all existing approaches to TER+E rely on traditional LMs. Importantly, figurative language explanation is not yet part of any downstream tasks on which LLMs have been recently tested (Chang et al., 2024). Therefore, the TER prediction and Explanation generation capabilities of LLMs in the presence of figurative language are still unexplored.

4 LLM-based Approaches

We envisage various LLM-based strategies to tackle TER+E in the presence of figurative language. First, we consider a classical Zero-Shot Learning (ZSL) setting, where we test the inherent understanding and generative capabilities of various pre-trained LLMs, open-source and proprietary. Secondly, we prompt LLMs with few training examples which are representative of the different TER labels and figurative language types, i.e., the Few-Shot Learning (FSL) setting. Thirdly, we explore the use of LLMs with Chain-of-Thought (CoT) prompting leveraging an external model, similar to Gu et al. (2022b) for TER+E. Finally, we envisage the use of LightWeight Fine-Tuning (LWFT) to specialize the LLMs' knowledge on the addressed prediction and explanation tasks.

We tune the LLM prompts separately for the ZSL/FSL and CoT settings. To identify the optimal prompt per model and setting, we first define a variety of templates, both generic and more specialized, and then run a tuning stage on the validation set. Finally, the most effective prompts are selected based on the acc@60 metric, which is representative of both TER and Explanation performance (see Section 5.4). Due to the lack of space, a detailed report of the prompt tuning settings and corresponding results are given in Appendix B.

4.1 Zero-Shot and Few-Shot Learning

ZSL entails prompting the LLM with a natural language question without providing any additional examples. In this scenario, LLM generation exclusively relies on the capabilities of the pre-trained model. Conversely, in a FSL setting the LLM

prompt also includes few examples to guide the LLM's reasoning and text generation.

To tackle TER+E we provide human-curated examples of premise-hypothesis sentence pairs including figurative language. For each example, the prompt also contains the actual label (E or C) as well as the figurative language explanation. Prompting ad hoc examples has the twofold aim to support the LLM in effectively generating TER labels and explanations and to force the model to comply with a predefined output format. In these experiments we pick k in-context examples from the training instances by adopting the following three strategies.

Random: The examples are randomly picked from the training set by disregarding labels' distribution.

Balanced: The examples are uniformly sampled from the per-type instance sets, disregarding the actual label distribution in the training data.

Stratified: The examples are stratified over the training instances to preserve the original figurative language type distribution (see Table 5).

For all strategies when $k > 1$ we provide at least one example per label to avoid introducing bias in the in-context learning procedure.

4.2 Chain-of-Thought Prompting

We explore CoT as an alternative strategy of LLM prompting. CoT prompting leverages external models to prompt LLMs with additional contextual information (Wei et al., 2022). CoT has shown to improve LLMs ability to perform complex reasoning by making the intermediate reasoning steps explicit in the LLM prompt. The idea is to foster LLMs to mimic the reasoning steps in the generative process. Our CoT method employs the DREAM scene elaboration model (Gu et al., 2022a) to enrich the input premise-hypothesis sentence pairs. It produces an elaboration of the input situation according to four different dimensions: *emotion*, *motivation*, *consequence*, and *social norm*. The objective is to provide LLMs with scene elaborations of the input pairs that can serve as intermediate reasoning steps, potentially supporting the model in understanding the relation between the input sentences that include figurative language. Note that, similar to DREAM-FLUTE (Gu et al., 2022b), CoT in LLMs requires external knowledge. Thus, unlike other LLM-based methods such as ZSL, FSL, and LWFT, CoT is neither general-purpose nor easily portable to different scenarios.

For each premise-hypothesis pair, we separately generate the four possible scene elaborations using DREAM. As an example, given the premise "*I will sleep deeply and soundly tonight.*", the scene elaboration for the *motivation* dimension is "*I (myself)'s motivation is to get to bed.*". Similar to the ZSL and FSL settings, we prompt the LLM with the input sample and k additional examples. Both the premises and the hypotheses that constitute the input instance and each of the k examples, if any, are enriched with the scene elaboration information produced by DREAM. Specifically, we consider both one dimension at a time (i.e., augmenting each premise and hypothesis with only the corresponding emotion/motivation/consequence/social norm elaboration) and the combination of all four dimensions. Consider, as an example, the *motivation* dimension. For both the k examples and the current test instance, we enrich the premise-hypothesis pairs in the prompt as follows:

```
Premise: [P]
Premise (motivation): [P_motivation]
Hypothesis: [H]
Hypothesis (motivation): [H_motivation]
```

where we replace `[P]`/`[H]` with the premise and hypothesis, and `[P_motivation]`/`[H_motivation]` with the corresponding scene elaborations produced by DREAM.

4.3 LightWeight Fine-Tuning

To ensure a higher level of model specialization than the simple in-context learning, we adopt LightWeight LLM Fine-Tuning to address TER+E. The idea behind it is that providing few training examples can be not sufficient to capture figurative language properties. Hence, we specialize the LLMs' knowledge on the specific TER+E task. Previous research (Bigoulaeva et al., 2022; Gu et al., 2022b) has demonstrated the effectiveness of fine-tuning in achieving a good balance between computational effort and model specialization. Hence, we adopt this established strategy as an alternative method to tailor LLMs to the figurative language understanding and explanation tasks. We fine-tune LLMs using the standard causal language modeling objective. To balance computational complexity and model performance, we do not perform a full fine-tuning of all model parameters but update only a fraction of them. More details on model fine-tuning are reported in Section 5.3.

5 Experiments

We run an extensive set of experiments on a machine equipped with Intel® Core™ i9-10980XE CPU, 1 × NVIDIA® RTX A6000 48GB GPU, 128 GB of RAM running Ubuntu 22.04 LTS.

In the following, we present the dataset, the models, the experimental settings, and the evaluation metrics used throughout the experiments. Next, we discuss the main empirical results. Due to the lack of space, qualitative examples of model outputs are given in Appendix E.

5.1 Dataset

We run experiments on the FLUTE benchmark dataset (Chakrabarty et al., 2022). It consists of almost 9,000 natural language instances, partitioned into training (around 7,500) and test (approximately 1,500) sets. Each instance is a premise-hypothesis sentence pair labeled with the TER label (entailment or contradiction) and enriched with a natural language explanation.

The dataset includes four types of figurative language, i.e., sarcasm, metaphors, similes, and idioms. All the figurative expressions appear in the hypothesis sentences except for sarcasm, where the hypothesis might be literal as well. Similar to Gu et al. (2022b), we employ an 80-20 dataset split, stratified over the figurative language types, to partition the initial training set into training and validation sets. Sarcasm instances are the most frequent ones whereas all the other types of figurative language have similar frequency counts. Entailment and contradiction instances are roughly balanced (more detailed statistics in Appendix A).

5.2 Models

LLMs. We consider the following open-source LLMs: Llama2 (base and chat versions) (Touvron et al., 2023), Llama3.1 (base and instruction-tuned versions) (Llama Team, 2024), Mistral0.1 and Mistral0.3 (base and instruction-tuned versions) (Jiang et al., 2023), Gemma and Gemma2 (base and instruction-tuned versions) (Gemma Team, 2024a,b), and Zephyr (base version) (Tunstall et al., 2023).

Competitors. We consider the best-performing methods according to the FigLang 2022 Shared Task report (Saakyan et al., 2022), i.e., DREAM-FLUTE (Gu et al., 2022b) (all the 5 systems) and Cross-Task (Bigoulaeva et al., 2022) (we test both the Sequential Fine-Tuning model (SFT) and the hierarchical feature pipeline architecture leveraging multi-task learning, namely HiFeatMTL).

Baselines. As TER+E baseline we consider the T5 model presented in Chakrabarty et al. (2022). As TER-only baseline, we also test the RoBERTa Transformer model (Liu et al., 2019). For all open-source baselines we rely on the Hugging Face Transformers library. Finally, as proprietary LLMs we test GPT-3.5 Turbo (`gpt-3.5-turbo-0125`) (OpenAI, 2024a) and GPT-4o (`gpt-4o-2024-05-13`) (OpenAI, 2024b) models using the OpenAI API. We conduct experiments with GPT-3.5 Turbo and GPT-4o in both ZSL/FSL and CoT settings.

To ensure a fair comparison, we recomputed the results of the baseline methods and competitors using our own evaluation scripts. Specifically, for both FLUTE and Cross-Task, we used the outputs provided by the authors, while for DREAM-FLUTE, we reproduced the model outputs of the different systems using the official checkpoints.

5.3 Experimental settings

In all experiments we set the model's temperature hyperparameter to 0.0 and the LLMs' prompts to their tuned version. While employing CoT prompting, the premise and hypothesis scene elaborations are generated using the official DREAM checkpoint for the selected dimension(s). In the FSL setting, we consider the following number of examples: $k = \{1, 3, 5, 10, 20\}$ for the Random strategy, $k = \{8, 16, 24\}$ for the Balanced strategy, and $k = \{12, 18, 24\}$ for the Stratified strategy. When adopting CoT prompting, we use $k = 5$ examples. For LLM LWFT, we utilized Low-Rank Adaptation (Hu et al., 2022) with PEFT (Mangrulkar et al., 2022) to limit computational complexity and train less than 1% of the total number of parameters. We use AdamW optimizer and set the maximum learning rate to $lr = \{5 \cdot 10^{-5}, 7 \cdot 10^{-5}\}$, which is updated using a linear scheduler after a warmup phase of the 10% of the total number of training steps. We set the maximum input length to 256 tokens and use a batch size of 2. We fine-tune models for 5 epochs and select the model checkpoint based on the best validation loss.

5.4 Evaluation Metrics

We use both established evaluation metrics for text generation and the official FLUTE benchmark evaluation suite. To evaluate Explana-

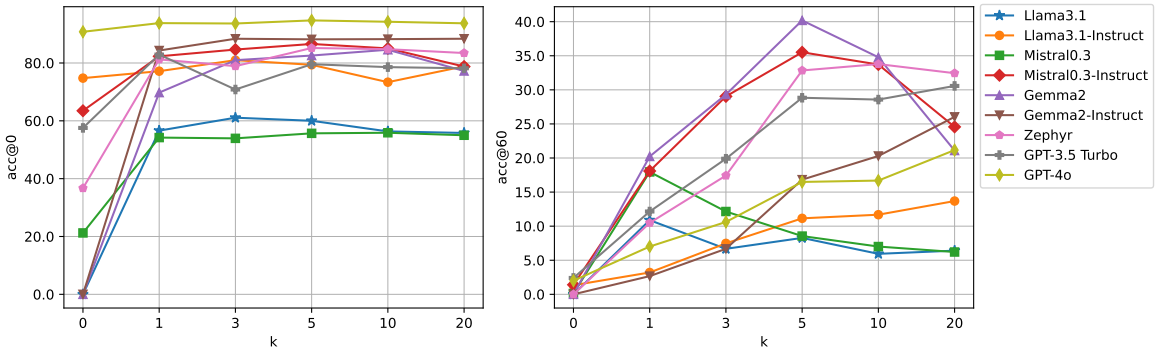

Figure 1: acc@0 (left) and acc@60 (right) results in the ZSL/FSL settings by varying number k of examples.

tion performance, we consider: (1) The standard syntax-oriented ROUGE metrics (Lin, 2004), i.e., ROUGE-1/2/L (R1/2/L) F1-scores, which measure the n-gram overlap between generated and reference explanations; (2) The semantic-based metrics BERTScore (BS) F1-score (Zhang et al., 2020) and BLEURT (BL) (Sellam et al., 2020), which has been shown to correlate better with human judgements (Chakrabarty et al., 2022).[2]

To evaluate TER-only performance, we adopt the standard classification accuracy, namely **acc@0**, which measures the percentage of correctly labeled instances. To evaluate the performance of the combined task (TER+E), we use the ad hoc metrics **acc@50** and **acc@60**, as defined in Chakrabarty et al. (2022). They measure the percentage of correctly labeled instances having an explanation score equal to or greater than 50 and 60, respectively, where the score is defined as the mean of BERTScore and BLEURT.

5.5 Results Overview

Zero and Few-Shot Learning. The plots in Figure 1 respectively compare the acc@0 and acc@60 scores achieved by different LLMs while varying the number of input examples from $k = 0$ (ZSL) to $k \in [1, 20]$ (FSL). For visualization purposes, we include only the results of the most recent LLMs, as previous versions show similar trends but generally achieve lower performance. In the ZSL setting, only Llama3.1-Instruct and GPT-4o achieve a TER acc@0 score above 70%. TER performance significantly improves while prompting LLMs with a few examples. Notably, Gemma2-Instruct, Mistral0.3-Instruct, and Zephyr turn out to be more accurate than the proprietary GPT-3.5 Turbo, with accuracy values close to 80%. However, GPT-4o achieves the best acc@0 performance in both ZSL and FSL settings. For most LLMs, the accuracy improvement yielded by FSL becomes roughly stable when feeding the LLM with more than 5 examples.

The acc@60 scores show similar trends (see the right-hand side plot in Figure 1), but the performance scores are halved compared to acc@0. The reason is that tackling TER+Explanation with a sufficiently high explanation score is significantly more challenging than addressing TER only. In the ZSL setting, almost all LLMs (including GPT-4o) obtain an acc@60 score close to zero because they are unable to generate reliable figurative language explanations. This confirms the inherent complexity of explaining figurative language. In the FSL setting (with $k > 5$) Gemma2, Mistral0.3-Instruct, and Zephyr show fairly good performance (acc@60 > 30). Specifically, Gemma2 yields the best acc@60 score (40.2) followed by Mistral0.3-Instruct and Zephyr. In terms of acc@0, open-source LLMs in the FSL setting yield performance comparable to or superior to that of GPT-3.5 Turbo and GPT-4o. However, in acc@60, open-source models consistently outperform proprietary ones. Notably, GPT-4o's acc@60 performance is significantly lower than that of other LLMs, including GPT-3.5 Turbo. By manually inspecting GPT-4o's outputs, we observe that, even while providing in-context examples, the generated explanations often tend to be a reformulation of the premise and hypothesis content separately rather than focusing on explaining the actual relation between the two sentences. Conversely, in the ZSL setting, GPT-4o achieves considerably better acc@0 results compared to open-source LLMs, likely due to its higher complexity and superior robustness.

We also empirically compare the strategies used to select the input examples in the FSL setting,

[2]BERTScore and BLEURT metrics are computed using the `microsoft/deberta-large-mnli` model and the recommended `BLEURT-20` checkpoint.

setting	acc@0	acc@50	acc@60	R1	R2	RL	BS	BL
Few-Shot Learning	**82.6**	**71.2**	**40.2**	**39.3**	**18.3**	**31.3**	**63.4**	**54.9**
CoT (emotion)	79.9	59.5	32.8	36.2	16.6	29.4	61.3	51.7
CoT (motivation)	79.2	63.0	*34.0*	*37.2*	*16.9*	*29.8*	*62.0*	*52.8*
CoT (consequence)	80.3	59.6	27.4	35.6	15.8	28.5	60.7	51.1
CoT (social norm)	79.9	*63.2*	30.7	35.8	15.6	28.6	61.4	52.0
CoT (all dimensions)	*82.2*	61.6	29.7	35.6	15.3	27.8	60.5	51.8

Table 1: CoT results using Gemma2. **Bold** and *italic* respectively denote the best and second-best score per metric.

i.e., Random, Balanced, and Stratified. Stratified yields slightly better acc@0 and acc@60 results compared to Random and Balanced, likely due to its capability to better take into account the distribution of figurative language types. Specifically, Zephyr achieves the best acc@60 results, followed by Gemma2. Notably, both models demonstrate higher performance with fewer examples compared to the Random strategy (e.g., Zephyr: 32.4 for $k = 20$ Random vs. 40.4 for $k = 18$ Stratified). The results improve further for $k = 24$, with Zephyr reaching an acc@60 score of 42.9.

Chain-of-Thought prompting. Table 1 summarizes the results of the best-performing LLM (i.e., Gemma2) with CoT prompting. CoT does not appear to be beneficial for TER (acc@0 of the best CoT method 82.2 vs. FSL 82.6). Similarly, for TER+E the external knowledge prompted via CoT proves to be unhelpful for LLMs (e.g., acc@60 CoT 34.0 vs. FSL 40.2). This is likely because DREAM elaborations are either not complementary to or not sufficiently relevant compared to the pre-trained knowledge of the LLM. Similar results, not reported here for the sake of brevity, hold for the other LLMs.

LightWeight Fine-Tuning. Table 2 reports the results achieved by LLMs with LWFT and a comparison with the baselines and state-of-the-art models. For the sake of brevity, we report only the best configuration results for GPT-3.5 Turbo and GPT-4o, i.e., FSL with $k = 20$ and $k = 24$ examples, respectively. Focusing on the TER-only task, most LLMs (except for Llama2) perform better than or as well as the state-of-the-art in terms of acc@0. Specifically, Gemma2-Instruct achieves the highest acc@0 score (97.0), followed by Mistral0.2-Instruct (96.4). Among the tested competitors, System 4 of DREAM-FLUTE (classify-then-explain using T5 (Raffel et al., 2020)) achieves the best score (95.2), though lower than Gemma2-Instruct. The ensemble approach (DREAM-FLUTE System 5) achieves results comparable to LLMs. However, as it relies on a mix of seven different models, its strategy and usability are not directly comparable with LLM-based methods.

Considering the TER+E task, LLMs with LWFT achieve significantly better performance than all existing approaches in terms of both acc@50 and acc@60. Considering acc@60, the best-performing models are Gemma2 and Gemma2-Instruct, with scores of 72.8 and 72.5, respectively. Among competitors, the best results are achieved by the SFT approach of Cross-Task and System 4 of DREAM-FLUTE, with acc@60 scores of 63.1 and 61.3, respectively. Consequently, our best model (i.e., Gemma2) shows an improvement of +9.7 and +11.5 points. Notably, the performance gap between LLMs and competitors further increases as we move from the acc@0 score (TER-only) to the acc@50 and acc@60 metrics (TER+Explanation).

The Explanation-only performance scores (ROUGE, BERTScore, BLEURT) confirm the expectation. Apart from Llama3.1-Instruct, all the tested LLMs demonstrate superior performance in generating natural language explanations compared to previous approaches. Given the surprisingly low acc@60 score of Llama3.1-Instruct, we checked its explanation scores and found that in many cases they are slightly lower than 60, which explains the sharp decline in the acc@60 metric. Our models perform best even when considering the ROUGE-1/2/L, BERTScore and BLEURT metrics (+1.4/0.4/0.7, +1.6 and +2.6 points, respectively). The proprietary GPT-3.5 Turbo and GPT-4o models with ZSL and FSL perform significantly worse than fine-tuned LLMs, confirming the need of a dedicated model fine-tuning.

We also conducted an additional analysis of the generated explanations. The results are available in Appendix C.

Performance per figurative language type and label. We analyze the variations in performance between different types of figurative language. Table 3 reports the acc@0 and acc@60 scores for

	model	acc@0	acc@50	acc@60	R1	R2	RL	BS	BL
FLUTE (Chakrabarty et al., 2022)	T5	81.7	74.9	48.5	42.2	19.9	34.4	66.0	56.3
DREAM-FLUTE (Gu et al., 2022b)	System1	94.6	86.7	56.6	46.1	24.8	39.3	67.3	57.1
	System2	94.8	86.9	56.9	46.3	24.8	39.5	67.3	56.9
	System3 (emotion)	93.9	87.6	57.2	46.2	24.9	39.4	67.5	57.2
	System3 (motivation)	94.8	87.2	55.8	45.9	24.7	39.1	67.3	56.8
	System3 (consequence)	94.6	87.3	58.6	46.5	25.1	39.6	67.4	57.1
	System3 (social norm)	92.4	84.9	56.1	45.4	24.0	38.7	67.1	56.9
	System3 (all dimensions)	94.9	87.7	56.9	46.0	24.8	39.2	67.5	56.9
	System4	95.2	89.9	61.3	46.9	25.5	39.9	68.5	58.7
	System5 (ensemble)	96.2	88.5	59.1	46.5	25.1	39.6	67.4	57.1
Cross-Task (Bigoulaeva et al., 2022)	SFT	92.2	87.5	63.1	47.1	26.7	40.6	68.8	59.4
	HiFeatMTL	94.8	86.7	55.3	45.9	25.8	39.3	67.6	58.2
GPT-3.5 Turbo (OpenAI, 2024a)	Few-Shot Learning	78.1	65.4	30.6	34.6	13.7	25.4	61.1	52.7
GPT-4o (OpenAI, 2024b)	Few-Shot Learning	94.7	74.9	28.2	33.1	13.3	24.5	59.1	52.6
GPT-as-an-Expert	Gemma2 + GPT-4o	95.4	91.3	72.6	48.3	26.9	41.0	70.2	61.7
Ours (Fine-Tuning)	Llama2 7B	93.8	90.7	68.3	46.8	25.3	39.5	68.8	60.9
	Llama2-Chat 7B	94.8	91.1	69.9	47.1	25.6	39.8	69.4	61.1
	Llama3.1 8B	95.7	91.8	70.6	48.0	26.3	40.6	70.0	61.3
	Llama3.1-Instruct 8B	95.9	89.1	43.5	34.0	18.1	28.3	58.0	61.2
	Mistral0.1 7B	96.0	92.3	70.3	47.1	25.8	39.6	68.9	61.6
	Mistral0.2-Instruct 7B	96.4	93.7	71.1	47.3	25.5	39.6	69.0	61.6
	Mistral0.3 7B	96.0	92.5	72.2	48.0	26.4	40.5	70.0	61.6
	Mistral0.3-Instruct 7B	96.2	92.5	72.1	47.8	26.5	40.7	70.2	61.7
	Gemma 7B	95.1	90.3	63.6	45.0	22.4	36.7	67.5	58.9
	Gemma-Instruct 7B	95.1	85.0	56.0	41.9	19.2	34.0	65.7	56.6
	Gemma2 9B	95.9	91.6	**72.8**	**48.5**	**27.1**	**41.3**	**70.4**	61.7
	Gemma2-Instruct 9B	**97.0**	**94.9**	72.5	48.0	26.8	40.4	70.0	**62.0**
	Zephyr 7B	96.2	92.6	72.1	47.8	26.2	40.3	69.7	61.8

Table 2: Results on the FLUTE benchmark dataset. **Bold** denotes the best score for each metric.

model	acc@0					acc@60				
	sarcasm	metaphor	simile	idiom	overall	sarcasm	metaphor	simile	idiom	overall
RoBERTa-MNLI	89.8	88.7	70.4	90.8	86.5	–	–	–	–	–
FLUTE (Chakrabarty et al., 2022) – T5	91.6	73.3	62.8	79.2	81.7	56.2	24.2	31.2	66.8	48.5
DREAM-FLUTE (Gu et al., 2022b) – System4	97.2	**94.3**	90.0	95.2	95.2	64.9	45.9	50.4	76.4	61.3
Cross-Task (Bigoulaeva et al., 2022) – SFT	95.4	89.1	84.8	93.2	92.2	67.6	47.9	49.2	78.8	63.1
GPT-3.5 Turbo – 20-shot (Random)	75.7	83.0	70.4	88.4	78.1	32.9	25.4	19.6	39.6	30.6
GPT-4o – 24-shot (Stratified)	96.2	93.1	92.0	94.4	94.7	29.8	20.1	27.2	32.4	28.2
Ours – Gemma2 9B	**97.3**	91.9	**94.4**	**97.2**	**95.9**	**78.4**	**51.6**	**62.8**	**87.2**	**72.8**

Table 3: acc@0 and acc@60 detailed results of best models by figurative expression type. **Bold** denotes the best score for each figurative expression type.

a selection of best-performing models, including the baseline RoBERTa Transformer fine-tuned on the MNLI dataset and then adapted to TER binary classification (we keep only the entailment or contradiction labels while ignoring the neutral label).

Results show that Gemma2 performs best in acc@0 on three out of four figurative expression types and excels across all of them in acc@60. The most significant improvement in acc@0 is observed in the simile class (i.e., +4.4 points). However, the improvements are more pronounced in acc@60. Specifically, Gemma2 exhibits enhancements of +10.8, +3.7, +12.4, and +8.4 points in sarcasm, metaphor, simile, and idiom classes, respectively. Surprisingly, the RoBERTa baseline achieves an overall acc@0 score even higher than FLUTE T5 and GPT-3.5 Turbo models. However, previous works have shown that correctly predicting the TER label alone is not sufficient but it is also necessary to provide natural language explanations for the label predictions (Camburu et al., 2018).

We also include detailed results per class in Table 4, where we provide the acc@0 and acc@60 scores of the best-performing models separately for the entailment and contradiction labels. Results indicate that Gemma2 achieves the highest performance in acc@0 on entailment instances, while it is slightly outperformed by DREAM-FLUTE on the contradiction class. More significant improvements are observed in acc@60, where Gemma2 surpasses the second-best-performing approach (i.e., Cross-Task) by +11.1 and +8.7 points for the entailment

model	acc@0			acc@60		
	entailment	contradiction	overall	entailment	contradiction	overall
RoBERTa-MNLI	88.1	85.4	86.5	–	–	–
FLUTE (Chakrabarty et al., 2022) – T5	73.0	88.6	81.7	44.8	51.4	48.5
DREAM-FLUTE (Gu et al., 2022b) – System4	92.4	**97.3**	95.2	58.5	63.4	61.3
Cross-Task (Bigoulaeva et al., 2022) – SFT	93.2	91.5	92.2	60.7	65.0	63.1
GPT-3.5 Turbo – 20-shot (Random)	58.6	93.6	78.1	22.3	37.1	30.6
GPT-4o – 24-shot (Stratified)	93.8	95.4	94.7	24.1	31.5	28.2
Ours – Gemma2 9B	**96.8**	95.2	**95.9**	**71.8**	**73.7**	**72.8**

Table 4: acc@0 and acc@60 detailed results of best models by TER label. **Bold** denotes the best score for each TER label.

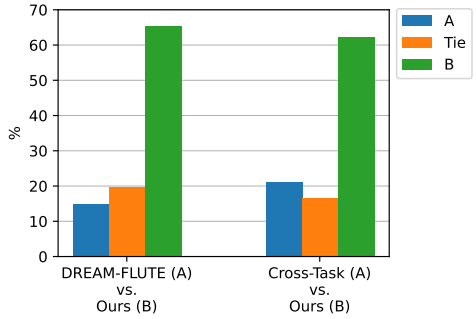

Figure 2: A/B test results. A is DREAM-FLUTE System 4 (left) and Cross-Task SFT (right), whereas B is our best-performing LLM (i.e., Gemma2).

and contradiction labels, respectively. These results highlight the superior performance of our model both across the different types of figurative expressions and TER labels.

A/B test. We complement the quantitative evaluation based on the standard FLUTE and text generation metrics (see Section 5.4) by conducting an A/B test supervised by the GPT-4o model (OpenAI, 2024b). Specifically, inspired by Zheng et al. (2023), for each test instance we ask GPT-4o to compare the TER+Explanation outputs of our best-performing model (Gemma2) with those of DREAM-FLUTE (System 4) and Cross-Task (SFT), and return whether the competitor's outcome is better than ours (option A), our model's output is better (option B), or the comparison ends in a draw. Figure 2 displays the A/B test results, showing that against both competitors, the outputs generated by our model are considered of better quality in nearly 65% of cases.

5.6 GPT-as-an-Expert

We also analyze GPT-4o performance *as an expert reviewer*, i.e., given a premise-hypothesis pair and the TER+E outputs of our best-performing model (i.e., Gemma2) we ask GPT-4o to check the outputs and, if need be, correct them. Based on our experiments, GPT-4o tends to rewrite explanations using its own style even when it is not necessary. Even if we provide GPT-4o with examples of ground truth explanations using in-context learning, it perseveres in unnecessarily modifying the already correct explanations. The results in Table 2 show that GPT-as-an-Expert performs worse than Gemma2 across all metrics. GPT-4o never modifies the label alone, corrects only the explanation in 121 cases, both in 48 cases, and neither in the remaining 1329 cases. To be more conservative, we also try to update our model's predicted label and explanation only when GPT-4o predicts a different label. However, the number of incorrect GPT-4o edits (27) is still comparable to those of correct ones (21). Further details are given in Appendix D.

6 Conclusions and Future Work

We studied LLM-based solutions to TER prediction and Explanation in the presence of figurative language. Fine-tuned LLMs outperform all state-of-the-art models suited to TER+E. In-context learning turns out to beneficial, compared to ZSL, while feeding LLMs with few examples (from 5 to 10), whereas CoT prompting using external knowledge turns out to be unhelpful. LWFT consistently and significantly improves LLMs performance on Explanation. Although closed-source models exhibit strong TER prediction performance, they unexpectedly perform worse than smaller, open-source models (e.g., Gemma2 9B) on Explanation.

As future work, we aim to address inconsistencies in TER+E outcomes, e.g., when the explanation is valid but the prediction is wrong or vice versa. We also plan to develop fine-tuned LLMs tailored to each type of figurative language and provide a compendium of error patterns that could be useful for LLM instruction tuning. Finally, we would like to analyze multimodal extensions of the TER+Explanation task (Saakyan et al., 2024).

Limitations

We identify the following limitations of our work:

- The FLUTE benchmark dataset contains only English sentence pairs, therefore we did not assess the figurative language understanding capabilities of LLMs in other languages. Expanding this analysis could improve the generalizability and portability of our results.
- Our analysis was limited to the types of figurative language included in the FLUTE benchmark dataset (i.e., sarcasm, metaphor, simile, and idiom). However, additional types of figurative expressions exist (e.g., personification, hyperbole, irony), which the trained LLMs might not be able to handle correctly.
- While LLMs have shown promising results for understanding and explaining figurative language in sentence pairs, real-world use cases may involve more complex scenarios, requiring the model to handle longer and more nuanced contextual information.
- LLM prompt tuning is known to be costly and prone to errors. Finding the best compromise between performance optimization and computational efficiency is out of the scope of the present work.
- LLM Fine-Tuning is computationally intensive and requires ad hoc hardware, even while implementing LightWeight Fine-Tuning. This could limit the reproducibility of the results.
- Due to computational constraints, we limited our analysis to small-sized open-source models (i.e., up to 9 billion parameters). We acknowledge that larger models could further enhance performance, although we believe that our results provide a reasonable balance between performance and computational demands.

Ethical Considerations

Generative AI is potentially harmful as could produce offensive, biased, or fake content. Therefore, their use in complex scenarios involving nuanced and non-literal expressions such as figurative language should be made with caution to avoid harm and spreading misinformation. LMs are also known to suffer from hallucination and language bias, potentially ignoring or misunderstanding the meaning of figurative language expressions used in dialects and less spoken languages. Additionally, figurative language can vary significantly across cultures, so models should recognize and respect cultural differences without reinforcing stereotypes or misrepresenting cultural nuances. Furthermore, some figurative language expressions may involve sensitive or potentially triggering content, and models should handle such content with care.

The data and models used in the experiments are public and not under our control. The use of LLMs must be sustainable and avoid unnecessary, energy-intensive training. In our experiments, we prioritized reusing pre-trained models, checkpoints, and outputs whenever possible to avoid wasting resources.

Acknowledgments

The work by Giuseppe Gallipoli was carried out within the MICS (Made in Italy – Circular and Sustainable) Extended Partnership and received funding from Next-GenerationEU (Italian PNRR – M4 C2, Invest 1.3 – D.D. 1551.11-10-2022, PE00000004). This study was also partially carried out within the FAIR (Future Artificial Intelligence Research) and received funding from Next-GenerationEU (Italian PNRR – M4 C2, Invest 1.3 – D.D. 1555.11-10-2022, PE00000013). This manuscript reflects only the authors' views and opinions, neither the European Union nor the European Commission can be considered responsible for them.

References

Rodrigo Agerri. 2008. Metaphor in textual entailment. In *Coling 2008: Companion volume: Posters*, pages 3–6, Manchester, UK. Coling 2008 Organizing Committee.

Rodrigo Agerri, John Barnden, Mark Lee, and Alan Wallington. 2008. Textual entailment as an evaluation framework for metaphor resolution: A proposal. In *Semantics in Text Processing. STEP 2008 Conference Proceedings*, pages 357–363. College Publications.

Irina Bigoulaeva, Rachneet Singh Sachdeva, Harish Tayyar Madabushi, Aline Villavicencio, and Iryna Gurevych. 2022. Effective cross-task transfer learning for explainable natural language inference with t5. In *Proceedings of the 3rd Workshop on Figurative Language Processing (FLP)*, pages 54–60, Abu Dhabi, United Arab Emirates (Hybrid). Association for Computational Linguistics.

Oana-Maria Camburu, Tim Rocktäschel, Thomas Lukasiewicz, and Phil Blunsom. 2018. e-snli: natural language inference with natural language explanations. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, NIPS'18, page 9560–9572, Red Hook, NY, USA. Curran Associates Inc.

Tuhin Chakrabarty, Debanjan Ghosh, Adam Poliak, and Smaranda Muresan. 2021. Figurative language in recognizing textual entailment. In *Findings of the Association for Computational Linguistics: ACL/IJCNLP 2021, Online Event, August 1-6, 2021*, volume ACL/IJCNLP 2021 of *Findings of ACL*, pages 3354–3361. Association for Computational Linguistics.

Tuhin Chakrabarty, Arkadiy Saakyan, Debanjan Ghosh, and Smaranda Muresan. 2022. FLUTE: Figurative language understanding through textual explanations. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 7139–7159, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Yupeng Chang, Xu Wang, Jindong Wang, Yuan Wu, Linyi Yang, Kaijie Zhu, Hao Chen, Xiaoyuan Yi, Cunxiang Wang, Yidong Wang, Wei Ye, Yue Zhang, Yi Chang, Philip S. Yu, Qiang Yang, and Xing Xie. 2024. A survey on evaluation of large language models. *ACM Trans. Intell. Syst. Technol.*, 15(3).

Gemma Team. 2024a. Gemma: Open Models Based on Gemini Research and Technology. *Preprint*, arXiv:2403.08295.

Gemma Team. 2024b. Gemma2: Improving Open Language Models at a Practical Size. *Preprint*, arXiv:2408.00118.

Maarten Grootendorst. 2020. Minimal keyword extraction with BERT.

Yuling Gu, Bhavana Dalvi, and Peter Clark. 2022a. DREAM: Improving situational QA by first elaborating the situation. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1115–1127, Seattle, United States. Association for Computational Linguistics.

Yuling Gu, Yao Fu, Valentina Pyatkin, Ian Magnusson, Bhavana Dalvi Mishra, and Peter Clark. 2022b. Just-DREAM-about-it: Figurative language understanding with DREAM-FLUTE. In *Proceedings of the 3rd Workshop on Figurative Language Processing (FLP)*, pages 84–93, Abu Dhabi, United Arab Emirates (Hybrid). Association for Computational Linguistics.

Randal Holme. 2004. *Using Figurative Language*, pages 28–58. Palgrave Macmillan UK, London.

Edward J Hu, yelong shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*.

Harsh Jhamtani, Varun Gangal, Eduard H. Hovy, and Taylor Berg-Kirkpatrick. 2021. Investigating robustness of dialog models to popular figurative language constructs. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7-11 November, 2021*, pages 7476–7485. Association for Computational Linguistics.

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. *Preprint*, arXiv:2310.06825.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized BERT pretraining approach. *CoRR*, abs/1907.11692.

Llama Team. 2024. The Llama 3 Herd of Models. *Preprint*, arXiv:2407.21783.

Sourab Mangrulkar, Sylvain Gugger, Lysandre Debut, Younes Belkada, Sayak Paul, and Benjamin Bossan. 2022. Peft: State-of-the-art parameter-efficient fine-tuning methods. `https://github.com/huggingface/peft`.

OpenAI. 2024a. OpenAI GPT-3.5.

OpenAI. 2024b. OpenAI GPT-4o.

Dylan Phelps, Thomas M. R. Pickard, Maggie Mi, Edward Gow-Smith, and Aline Villavicencio. 2024. Sign of the times: Evaluating the use of large language models for idiomaticity detection. In *Proceedings of the Joint Workshop on Multiword Expressions and Universal Dependencies (MWE-UD) @ LREC-COLING 2024*, pages 178–187, Torino, Italia. ELRA and ICCL.

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. *J. Mach. Learn. Res.*, 21(1).

Arkadiy Saakyan, Tuhin Chakrabarty, Debanjan Ghosh, and Smaranda Muresan. 2022. A report on the FigLang 2022 shared task on understanding figurative language. In *Proceedings of the 3rd Workshop on Figurative Language Processing (FLP)*, pages 178–183, Abu Dhabi, United Arab Emirates (Hybrid). Association for Computational Linguistics.

Arkadiy Saakyan, Shreyas Kulkarni, Tuhin Chakrabarty, and Smaranda Muresan. 2024. V-flute: Visual figurative language understanding with textual explanations. *Preprint*, arXiv:2405.01474.

Thibault Sellam, Dipanjan Das, and Ankur Parikh. 2020. BLEURT: Learning robust metrics for text generation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7881–7892, Online. Association for Computational Linguistics.

Kevin Stowe, Prasetya Utama, and Iryna Gurevych. 2022. IMPLI: Investigating NLI models' performance on figurative language. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 5375–5388, Dublin, Ireland. Association for Computational Linguistics.

Hugo Touvron et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *CoRR*, abs/2307.09288.

Lewis Tunstall, Edward Beeching, Nathan Lambert, Nazneen Rajani, Kashif Rasul, Younes Belkada, Shengyi Huang, Leandro von Werra, Clémentine Fourrier, Nathan Habib, Nathan Sarrazin, Omar Sanseviero, Alexander M. Rush, and Thomas Wolf. 2023. Zephyr: Direct distillation of lm alignment. *Preprint*, arXiv:2310.16944.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, brian ichter, Fei Xia, Ed Chi, Quoc V Le, and Denny Zhou. 2022. Chain-of-thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems*, volume 35, pages 24824–24837. Curran Associates, Inc.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020. BERTScore: Evaluating text generation with BERT. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, Hao Zhang, Joseph E Gonzalez, and Ion Stoica. 2023. Judging llm-as-a-judge with mt-bench and chatbot arena. In *Advances in Neural Information Processing Systems*, volume 36, pages 46595–46623. Curran Associates, Inc.

Appendices

In this supplementary material, we provide additional details as follows:

- Appendix A: FLUTE dataset statistics
- Appendix B: Prompt tuning
- Appendix C: Explanation analysis
- Appendix D: GPT-as-an-Expert
- Appendix E: Qualitative examples

A FLUTE dataset statistics

Table 5 reports the FLUTE benchmark dataset (Chakrabarty et al., 2022) statistics. We include the frequency distributions over both figurative language types and label values (entailment or contradiction) across the three dataset splits.

	# training	# validation	# test
sarcasm	2514	750	750
metaphor	998	250	248
simile	1000	250	250
idiom	1518	250	250
entailment	2551	632	663
contradiction	3479	868	835
total	6030	1500	1498

Table 5: FLUTE benchmark dataset (Chakrabarty et al., 2022) statistics.

B Prompt tuning

In the following, we include the different prompts we tested for both Zero/Few-Shot Learning and Chain-of-Thought settings. For the sake of readability, we report prompts 2-6 in the case where no examples are provided. If examples are provided, we repeat the prompt for each of the k examples, including the corresponding ground truth labels and explanations, as shown in prompt 1.

Zero- and Few-Shot Learning

Prompt 1

```
Is there a contradiction or entailment between
the premise and hypothesis?
Premise: [EX_Pj]
Hypothesis: [EX_Hj]
Label: [EX_Lj]
Explanation: [EX_Ej]
...
Is there a contradiction or entailment between
the premise and hypothesis?
Premise: [P]
Hypothesis: [H]
Label:
```

where we replace `[EX_Pj]`, `[EX_Hj]`, `[EX_Lj]` and `[EX_Ej]` with the premise, hypothesis, label and explanation for each of the k examples provided as input to the model, if any, and `[P]`/`[H]` with the premise and hypothesis of the current instance.

Prompt 2

```
I will provide you with a pair of sentences
consisting of a premise and a hypothesis containing
figurative language.
Is there a contradiction or entailment between
the premise and hypothesis?
Provide an explanation for your answer.
Premise: [P]
Hypothesis: [H]
Label:
```

Prompt 3

```
You are an expert in linguistics. I will provide
you with a pair of sentences (a premise and a
hypothesis) containing figurative language.
Your task is to determine whether the premise
entails or contradicts the hypothesis.
Provide an explanation for your answer.
Premise: [P]
Hypothesis: [H]
Label:
```

Prompts 1 and 2 proved to be the best-performing prompts across most models. We also use prompt 1 for model fine-tuning.

Chain-of-Thought

As an example, we consider to enrich each premise-hypothesis pair with the DREAM *motivation* dimension.

Prompt 4

```
Is there a contradiction or entailment between
the premise and hypothesis?
Premise: [P]
Premise (motivation): [P_motivation]
Hypothesis: [H]
Hypothesis (motivation): [H_motivation]
Label:
```

Prompt 5

```
I will provide you with a pair of sentences consisting
of a premise and a hypothesis containing figurative
language. For both the premise and the hypothesis,
I will provide you with additional motivation context.
Is there a contradiction or entailment between
the premise and hypothesis?
Provide an explanation for your answer.
Premise: [P]
Premise (motivation): [P_motivation]
Hypothesis: [H]
Hypothesis (motivation): [H_motivation]
Label:
```

model	ZSL and FSL			CoT		
	prompt 1	**prompt 2**	**prompt 3**	**prompt 4**	**prompt 5**	**prompt 6**
GPT-3.5 Turbo	**19.7**	16.5	14.2	**19.9**	16.3	16.2
GPT-4o	7.9	**9.1**	6.1	**2.6**	1.7	1.8
Llama2 7B	**1.5**	0.3	0.4	**0.7**	0.5	0.5
Llama2-Chat 7B	**9.5**	5.6	3.7	**12.8**	7.3	2.9
Llama3.1 8B	4.9	**5.9**	5.8	2.7	5.5	**8.1**
Llama3.1-Instruct 8B	**8.2**	7.4	5.0	4.5	3.7	**5.3**
Mistral0.1 7B	**10.6**	8.8	1.9	**17.9**	9.5	15.6
Mistral0.2-Instruct 7B	**17.1**	13.4	16.0	**15.7**	11.8	11.9
Mistral0.3 7B	10.5	**13.8**	12.5	6.8	**8.0**	7.6
Mistral0.3-Instruct 7B	27.4	**29.9**	27.9	**24.3**	23.7	24.2
Gemma 7B	**28.3**	7.1	10.1	**3.6**	2.9	3.3
Gemma-Instruct 7B	**12.5**	10.6	3.0	**15.6**	2.2	14.2
Gemma2 9B	27.5	27.3	**28.8**	30.4	30.9	**31.1**
Gemma2-Instruct 9B	**8.3**	5.6	4.9	6.8	6.0	**7.1**
Zephyr 7B	**17.2**	12.5	11.7	**20.3**	19.9	19.0

Table 6: acc@60 prompt tuning results on the validation set. **Bold** denotes the best prompt for each model, separately for ZSL/FSL and CoT.

model	avg # tokens
ground truth	29.4 ± 24.7
DREAM-FLUTE (Gu et al., 2022b)	27.5 ± 22.9
Cross-Task (Bigoulaeva et al., 2022)	26.6 ± 21.3
GPT-4o (OpenAI, 2024b)	47.2 ± 32.8
Gemma2 (Ours)	28.4 ± 23.2

Table 7: Average number of tokens of generated explanations by the best models and GPT-4o.

Prompt 6

```
You are an expert in linguistics. I will provide you
with a pair of sentences (a premise and a hypothesis)
containing figurative language.
For both the premise and the hypothesis, I will
provide you with additional motivation context.
Considering the pair of sentences and the additional
context, your task is to determine whether
the premise entails or contradicts the hypothesis.
Provide an explanation for your answer.
Premise: [P]
Premise (motivation): [P_motivation]
Hypothesis: [H]
Hypothesis (motivation): [H_motivation]
Label:
```

In this case, prompts 4 and 6 proved to be the best-performing prompts across most models.

Table 6 presents the acc@60 prompt tuning results on the validation set, which are used to determine the best prompt for each model, separately for ZSL/FSL and CoT settings.

C Explanation analysis

To provide further insights into the explanation generation results, we conduct an additional analysis comparing the explanations generated by the best models and GPT-4o with the ground truth, focusing on both average length and keyword overlap. Table 7 reports the average lengths of the generated explanations. We can observe that our fine-tuned best model Gemma2 shows the closest average length compared to the ground truth, followed by DREAM-FLUTE (System 4) and Cross-Task (SFT). In contrast, the length of GPT-4o's explanations deviates the most from the ground truth (i.e., 47.2 vs. 29.4) and exhibits the highest variance.

For keyword analysis, we extract keywords from the ground truth and the generated explanations using both TD-IDF and KeyBERT (Grootendorst, 2020). In the first case, the text is pre-processed (i.e., punctuation and stop word removal, lemmatization), while in the second case we rely on embeddings extracted using the `all-MiniLM-L6-v2` model. The idea is that, although ROUGE metrics already account for syntactic word overlap, they do not focus exclusively on keywords but rather on all words in the sentence (e.g., conjunctions, articles). After extracting the top-10 and top-K keywords from the ground truth and the generated explanations, respectively, we compare them using precision@K, recall@K, and an MRR-based metric.

Figure 3 displays the results for varying number of keywords $K \in \{1, 3, 5, 7, 10\}$. Considering the TF-IDF results, for higher values of K, precision and MRR decrease while recall increases. Our model consistently yields better performance across all metrics and values of K. Notably, GPT-4o always achieves significantly lower results (e.g.,

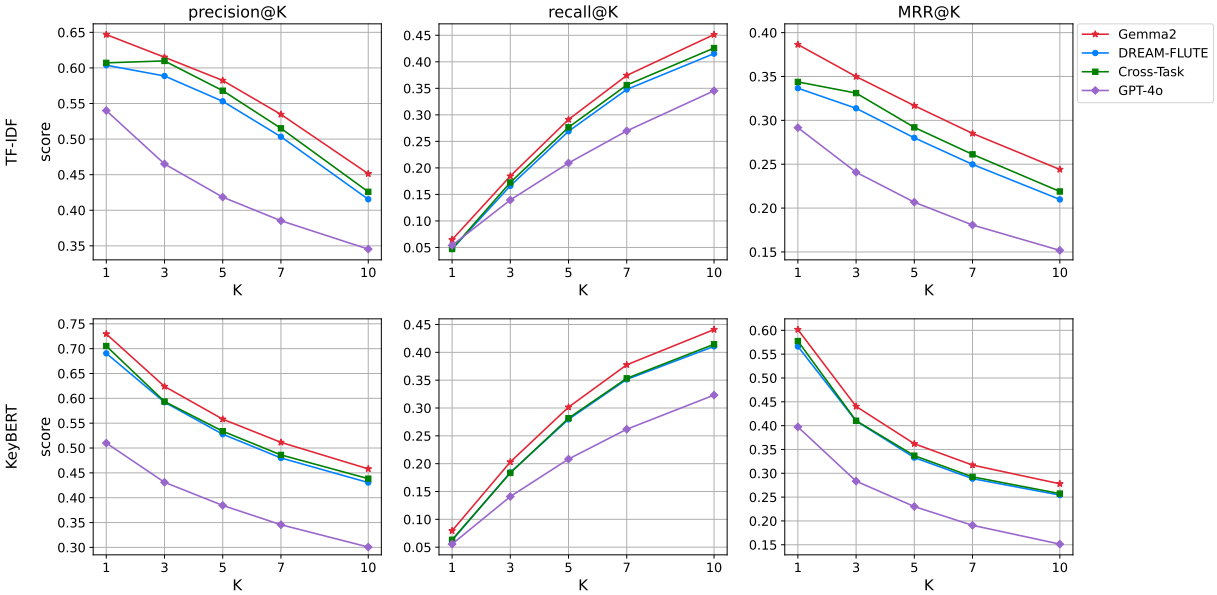

Figure 3: Precision, recall, and MRR for varying number of keywords K in the generated explanations compared to the ground truth by the best models and GPT-4o. Keywords are extracted using TF-IDF (first row) and KeyBERT (second row).

compared to Gemma2 for $K = 5$, 0.42 vs. 0.58 precision, 0.21 vs. 0.29 recall, and 0.21 vs. 0.32 MRR). KeyBERT results follow a similar trend: overall, they are higher than those obtained with TF-IDF, likely due to the more accurate and robust keyword extraction enabled by embedding representations. While DREAM-FLUTE and Cross-Task results are almost identical in this case, our model consistently achieves better performance. As with the TF-IDF results, GPT-4o shows the lowest performance in this case as well.

D GPT-as-an-Expert

Based on preliminary experiments, we use the following prompt to ask GPT-4o to check and, if necessary, correct or improve our best model's (i.e., Gemma2) predictions.

```
I will provide you with a pair of sentences consisting
of a premise and a hypothesis containing figurative
language. The task is to determine whether there is a
contradiction or entailment between the premise and
hypothesis, and provide an explanation for it.
I will also provide you with a model's prediction
("Entailment" or "Contradiction") and explanation.
Your task is to verify the correctness of the
prediction and, if needed, improve the explanation.
When modifying the explanation, do not explicitly
mention "premise" or "hypothesis", and keep the same
length and style of the model's generated one.

In the following you can find some examples of
explanations. Use the same style.
- [EX_Ej]
- ...

Premise: [P]
Hypothesis: [H]

Model's prediction
Label: [PRED_LABEL]
Explanation: [PRED_EXPL]

Answer with
Label:
Explanation:
If the label is the same as the model's prediction,
write "Correct". If the explanation does not need
improvement, write "Correct".
```

where we replace `[EX_Ej]` with the j-th explanation for each of the k examples provided to the model, if any, `[P]`/`[H]` with the premise and hypothesis, and `[PRED_LABEL]`/`[PRED_EXPL]` with our best model's corresponding prediction and explanation for the current instance.

Table 8 reports the results achieved by GPT-as-an-Expert when varying the number of explanations provided as examples to the model. Note that we include the results for both explanation update strategies, i.e., *always*, if we always update the explanation generated by our best model with the one revised by GPT-4o, and *label*, where we replace our model's explanation only when GPT-4o predicts a label different from the one originally predicted by our model. Considering the acc@60 metric, as expected, there is an increase in performance until reaching the maximum value for $k = 10$. Notably, the largest performance gap is observed between $k = 0$ and $k = 5$. This can be explained by the

# examples	explanation update	acc@0	acc@50	acc@60	R1	R2	RL	BS	BL
0	always	94.2	88.1	64.1	45.4	24.6	38.2	68.5	60.0
	label	94.2	90.1	71.7	48.0	26.6	40.7	70.0	61.4
5	always	94.9	90.5	69.2	47.0	25.9	39.8	69.5	61.0
	label	94.9	91.1	72.2	48.2	26.8	40.9	70.1	61.6
10	always	**95.4**	90.7	70.6	47.6	26.3	40.3	69.8	61.3
	label	**95.4**	**91.3**	**72.6**	**48.3**	**26.9**	**41.0**	**70.2**	**61.7**
20	always	95.0	90.3	69.8	47.5	26.2	40.1	69.7	61.3
	label	95.0	90.7	72.0	48.2	26.8	40.9	70.1	61.6

Table 8: GPT *as an expert* results for varying number of examples of explanations. **Bold** denotes the best score for each metric.

fact that, when no examples of explanations are provided, GPT-4o tends to rewrite the explanations produced by Gemma2 using its own style, which does not align well with the ground truth. Conversely, starting from $k = 5$, the style of the explanations revised by GPT-4o more closely matches the expected one. This is even more evident when comparing the results of the two explanation update strategies. Since GPT-4o often rewrites explanations using its own style, always replacing Gemma2's explanation with the one revised by GPT-4o results in lower performance compared to updating it only when the predicted label also changes. Specifically, for $k = 0$, the gap between the two strategies is particularly significant (e.g., 64.1 vs 71.7 acc@60), while it decreases when examples are provided. For $k > 0$, GPT-4o's explanations become more aligned with the ground truth, leading to fairly good results even in the *always* strategy.

Qualitative Results. We include in the following two examples, one success and one failure case, respectively, of outputs produced by using GPT-4o *as an expert reviewer*.

Example 1 (simile):
Premise: *It felt hard and dry.*
Hypothesis: *It felt like a portobello mushroom.*
Label: *Contradiction*
Explanation: *Portobello mushrooms are big and wet, so saying it felt like one would imply it felt slimy, not hard and dry*

Gemma2
Label: *Entailment*
Explanation: *A portobello mushroom is a type of mushroom that is known for being hard and dry.*

Gemma2 + GPT-4o
Label: *Contradiction*
Explanation: *A portobello mushroom is known for being soft and moist, not hard and dry.*

In the example above, GPT-4o effectively corrects Gemma2's predictions both in terms of label and explanation. In particular, unlike Gemma2, GPT-4o appears to be knowledgeable about the characteristics of Portobello mushrooms, likely due to its more extensive training data, and therefore correctly recognizes the contradiction between the premise and the hypothesis.

Example 2 (metaphor):
Premise: *The sun reflecting a dazzling light to us.*
Hypothesis: *The sun glared down on us.*
Label: *Entailment*
Explanation: *Glaring is used to mean that the sun is reflecting a strong light.*

Gemma2
Label: *Entailment*
Explanation: *The sun glared down on us means that the sun was shining brightly and was reflecting a lot of light.*

Gemma2 + GPT-4o
Label: *Contradiction*
Explanation: *The sun reflecting a dazzling light suggests a beautiful and pleasant scene, while the sun glaring down implies an intense, harsh, and possibly uncomfortable light.*

In the example above, GPT-4o modifies both the label and explanation predicted by Gemma2 leading to an incorrect final output. In this case, although both the phrase "dazzling light" in the premise and the verb "to glare" in the hypothesis refer to a very bright light, GPT-4o perceives the former as a pleasant scene while the latter as an uncomfortable situation. Consequently, it interprets the relation between the two sentences as a contradiction.

E Qualitative examples

In the following, we provide some examples of success and failure cases generated by our best-performing model (i.e., Gemma2), DREAM-FLUTE System 4 (Gu et al., 2022b), and Cross-Task SFT (Bigoulaeva et al., 2022). For each example, we also report the ground truth label and explanation to facilitate the qualitative assessment of the model outputs.

Example 3 (simile):
Premise: *The basement they shut him in looked safe and spacious.*
Hypothesis: *They had shut him in a basement that looked like a freight elevator.*
Label: *Contradiction*
Explanation: *A freight elevator is typically small and cramped, not safe and spacious.*

Cross-Task (SFT)
Label: *Entailment*
Explanation: *A freight elevator is a large, safe space, so the basement they shut him in must have looked safe and spacious.*

DREAM-FLUTE (System 4)
Label: *Entailment*
Explanation: *A freight elevator is a large, safe, and spacious space, so the basement they shut him in must have looked safe and spacious.*

Ours (Gemma2)
Label: *Contradiction*
Explanation: *A freight elevator is a small, cramped space, whereas a safe and spacious basement would be larger and more comfortable.*

In the example above, both Cross-Task and DREAM-FLUTE methods predict the wrong TER label. The error is then propagated in the generated explanations which, although coherent with the predicted label, are misleading. In contrast, our model accurately predicts the correct label (i.e., contradiction) and generates a high-quality explanation that aligns with the ground truth. Particularly noteworthy is the presence of the exact words "small and cramped" in both the gold reference and our model output.

Example 4 (idiom):
Premise: *She'd always seemed fragile, though according to Jess she was as susceptible to sickness and injury as a newborn baby and [...]*
Hypothesis: *She'd always seemed fragile, though according to Jess she was as tough as old boots and could face off the most dangerous badasses.*
Label: *Contradiction*
Explanation: *To be tough as old boots means to be resilient and not susceptible to sickness or injury, but this context describes the individual as being susceptible to sickness and injury.*

Cross-Task (SFT)
Label: *Contradiction*
Explanation: *Old boots are not tough, they are susceptible to sickness and injury*

DREAM-FLUTE (System 4)
Label: *Entailment*
Explanation: *Old boots are not very durable, so the simile is saying that she is not very tough.*

Ours (Gemma2)
Label: *Contradiction*
Explanation: *To be tough as old boots means to be very tough and durable, but in this context she is as susceptible to sickness and injury as a newborn baby.*

In this example, Cross-Task predicts the correct label (i.e., contradiction). However, the model fails to understand the meaning of the idiom "tough as old boots", thus producing a wrong explanation. Similarly, DREAM-FLUTE produces an explanation that conveys the opposite meaning of the intended one, alongside an incorrect label prediction. On the other hand, our model correctly recognizes the contradiction between the input premise and hypothesis. The label prediction is supported by a correct explanation, demonstrating the model's understanding of the idiom. Furthermore, the explanation closely matches the gold reference.

Example 5 (sarcasm):
Premise: *I had to take a test for work and didnt have time to study so I snuck notes in so I could get a high score.*
Hypothesis: *I had to take a test for work and didnt have time to study so I fine-tuned my performance by sneaking notes in so I could get a high score*
Label: *Contradiction*
Explanation: *Sneaking notes into a test is considered cheating and is not fine-tuning one's performance but rather taking the easy way.*

Cross-Task (SFT)
Label: *Entailment*
Explanation: *Fine-tuning one's performance means to improve it by taking notes, which is what the person did by sneaking notes in.*

DREAM-FLUTE (System 4)
Label: *Entailment*

Explanation: *It is often seen as cheating if someone is caught sneaking notes in a test and it is often seen as a way to get a better grade*

Ours (Gemma2)
Label: *Contradiction*
Explanation: *Sneaking notes in during a test is considered cheating and is not fine-tuning one's performance.*

In the example above, both Cross-Task and DREAM-FLUTE methods predict the wrong label. Although DREAM-FLUTE explanation mentions the verb "cheating", it does not recognize the sarcasm in the hypothesis. In contrast, our model both correctly predicts the label (i.e., contradiction) and explains the sarcasm of the situation, closely matching the ground truth.

Example 6 (metaphor):
Premise: *That huge guy beside him made him look so puny.*
Hypothesis: *His friend was very tall and he looked like a chess piece beside him.*
Label: *Entailment*
Explanation: *A chess piece is much smaller than the board it is on, so when one person is standing next to another who is much taller, they can look like a chess piece.*

Cross-Task (SFT)
Label: *Entailment*
Explanation: *A chess piece is very tall, so the simile is saying that the person next to him looks very tall.*

DREAM-FLUTE (System 4)
Label: *Contradiction*
Explanation: *A chess piece is a small, intricate piece of furniture, so it would not make sense for someone to look like one if they were tall and looked puny*

Ours (Gemma2)
Label: *Entailment*
Explanation: *A chess piece is a small object, so the person would look small and puny next to someone who is very tall.*

In this example, Cross-Task predicts the correct label (i.e., entailment); however, the generated explanation is incorrect as it conveys the opposite meaning of the ground truth. DREAM-FLUTE explanation is only partially correct, furthermore the predicted label is wrong. On the other hand, Gemma2 predicts the correct label and effectively explains the meaning of the metaphor in the generated explanation.

Example 7 (simile):
Premise: *She did not keep still*
Hypothesis: *She kept as still as a bouncing superball*
Label: *Entailment*
Explanation: *A superball is a ball that is constantly moving and bouncing, so the simile is saying that she was not still at all.*

Cross-Task (SFT)
Label: *Entailment*
Explanation: *A superball is a very fast moving object, so to say someone keeps as still as a superball would be to say that they do not keep still.*

DREAM-FLUTE (System 4)
Label: *Contradiction*
Explanation: *A bouncing superball is not still, so she did not keep still*

Ours (Gemma2)
Label: *Contradiction*
Explanation: *A superball is a toy that bounces when dropped, so saying someone is as still as a superball would mean that they are not still at all.*

In the example above, Cross-Task's predicted label and explanation are both correct, while DREAM-FLUTE predicts the wrong label but produces a correct, although concise, explanation. Similarly, although our model generates an explanation that closely aligns with the gold reference, it predicts the wrong label.

Example 8 (metaphor):
Premise: *This office is as lively as a bustling beehive.*
Hypothesis: *This office is buzzing with activity.*
Label: *Entailment*
Explanation: *A beehive is a place where bees live and buzz.*

Cross-Task (SFT)
Label: *Entailment*
Explanation: *A beehive is a structure that is full of activity, so the simile is saying that the office is full of activity.*

DREAM-FLUTE (System 4)
Label: *Contradiction*
Explanation: *A beehive is a structure where many bees live together and produce honey, so it would not be lively*

Ours (Gemma2)
Label: *Contradiction*
Explanation: *To be buzzing with activity means to*

be full of energy and activity, while to be as lively as a bustling beehive means to be full of life and activity.

In this example, similar to the previous case, CrossTask predicts the correct label (i.e., entailment) and generates a valid explanation. In contrast, DREAMFLUTE's predicted label and explanation are incorrect. Although Gemma2 correctly explains the meaning of the expressions "to be buzzing with activity" and "to be as lively as a bustling beehive", they are perceived as having contrasting meaning, resulting in a wrong label prediction.