Sentiment Analysis through LLM Negotiations

Anonymous ACL submission

Abstract

A standard paradigm for sentiment analysis is to rely on a singular LLM and makes the decision in a single round under the framework of in-context learning. This framework suffers the key disadvantage that the single-turn output generated by a single LLM might not deliver the perfect decision, just as humans sometimes need multiple attempts to get things right. This is especially true for the task of sentiment analysis where deep reasoning is required to address the complex linguistic phenomenon (e.g., clause composition, irony, etc) in the input.

To address this issue, this paper introduces a multi-LLM negotiation framework for sentiment analysis. The framework consists of a reasoning-infused generator to provide decision along with rationale, a explanation-deriving discriminator to evaluate the credibility of the generator. The generator and the discriminator iterate until a consensus is reached. The proposed framework naturally addressed the aforementioned challenge, as we are able to take the complementary abilities of two LLMs, have them use rationale to persuade each other for correction.

Experiments on a wide range of sentiment analysis benchmarks (SST-2, Movie Review, Twitter, yelp, amazon, IMDB) demonstrate the effectiveness of proposed approach: it consistently yields better performances than the ICL baseline across all benchmarks, and even superior performances to supervised baselines on the Twitter and movie review datasets.

1 Introduction

Sentiment analysis (Pang and Lee, 2008; Go et al., 2009; Maas et al., 2011a; Zhang and Liu, 2012; Baccianella et al., 2010; Medhat et al., 2014; Bakshi et al., 2016; Zhang et al., 2018) aims to extract opinion polarity expressed by a chunk of text. Recent advances in large language models (LLMs) (Brown et al., 2020; Ouyang et al., 2022; Touvron et al., 2023a,b; Anil et al., 2023; Zeng et al., 2022b; OpenAI, 2023; Bai et al., 2023) open a new door for the resolving the task (Lu et al., 2021; Kojima et al., 2022; Wang et al., 2022b; Wei et al., 2022b; Wan et al., 2023; Wang et al., 2023; Sun et al., 2023b,a; Lightman et al., 2023; Li et al., 2023; Schick et al., 2023): under the paradigm of in-context learning (ICL), LLMs are able to achieve performances comparable to supervised learning strategies (Lin et al., 2021; Sun et al., 2021; Phan and Ogunbona, 2020; Dai et al., 2021) with only a small number of training examples.

Existing approaches that harness LLMs for sentiment analysis usually rely on a **singular LLM**, and make a decision in a **single round** under ICL. This strategy suffers from the following disadvantage: the single-turn output generated by a single LLM might not deliver the perfect response: Just as humans sometimes need multiple attempts to get things right, it might take multiple rounds before an LLM makes the right decision. This is especially true for the task of sentiment analysis, where LLMs usually need to articulate the reasoning process to address the complex linguistic phenomenon (e.g., clause composition, irony, etc) in the input sentence.

To address the this issue, in this paper, we propose a multi-LLM negotiation strategy for sentiment analysis. The core of the proposed strategy is a generator-discriminator framework, where one LLM acts as the generator (G) to produce sentiment decisions, while the other acts as a discriminator (D), tasked with evaluating the credibility of the generated output from the first LLM. The proposed method innovates on three aspects: (1) Reasoning-infused generator (G): an LLM that adheres to a structured reasoning chain, enhancing the ICL of the generator while offering the discriminator the evidence and insights to evaluate its validity; (2) Explanation-deriving

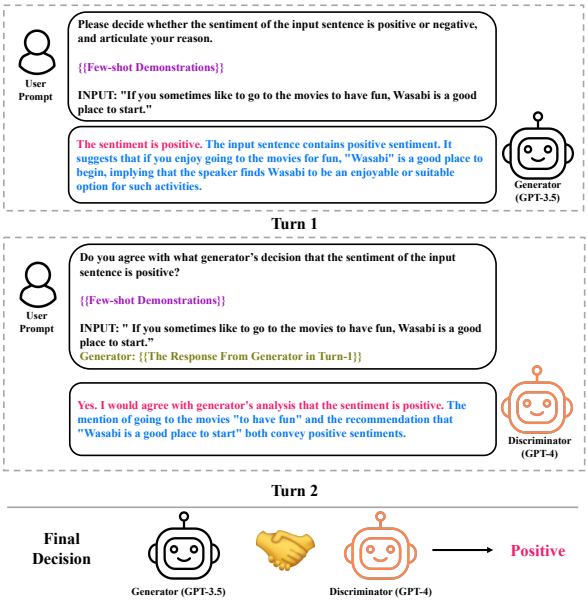

Figure 1: An illustration of a generator (G) and a discriminator (D) achieving consensus via a negotiation. Each round consists of a user prompt and a response from either G or D. Specifically, a user prompt includes four elements: a task description, few-shot demonstrations (abbreviate it for short), an input, and a response from the last turn (if applicable). Responses from G or D start with statements that the input contains positive sentiment, followed by rationale.

discriminator (D); other LLM designed to offer post-evaluation rationales for its judgments; (3) Negotiation: two LLMs act as the roles of the generator and the discriminator, and perform the negotiation until a consensus is reached.

This strategy harnesses the collective abilities of the two LLMs and provide with the channel for the model to correct imperfect responses, and thus naturally resolves the issue that a single LLM cannot yield the correct decision on its first try.

The contributions of this work can be summarized as follows: 1) we provide a novel perspective on how sentiment analysis can benefit from multi-LLM negotiation. 2) we introduce a Generator-Discriminator Role-switching Decision-Making framework that enables multi-LLM collaboration through iteratively generating and validating sentiment categorizations. 3) our empirical findings offer evidence for the efficacy of the proposed approach: experiments on a wide range of sentiment analysis benchmarks (SST-2, Movie Review, Twitter, yelp, amazon, IMDB) demonstrate that the proposed method consistently yields better performances than the ICL baseline across all benchmarks, and even superior performances to supervised baselines on the Twitter and movie review datasets.

2 Related Work

2.1 Sentiment Analysis

Sentiment analysis (Pang and Lee, 2008; Go et al., 2009; Maas et al., 2011a; Zhang and Liu, 2012; Baccianella et al., 2010; Medhat et al., 2014; Bakshi et al., 2016; Zhang et al., 2018) is a task that aims to determine the overall sentiment polarity (e.g., positive, negative, neutral) of a given text. Earlier work often formalized the task as a two-step problem: (1) extract features using RNNs (Socher et al., 2013; Qian et al., 2016; Peled and Reichart, 2017; Wang et al., 2016b; Guggilla et al., 2016; Vo and Zhang, 2015), CNNs (Kalchbrenner et al., 2014; Wang et al., 2016a; Guan et al., 2016; Yu and Jiang, 2016; Mishra et al., 2017), pretrained language models (Lin et al., 2021; Sun et al., 2021; Phan and Ogunbona, 2020; Dai et al., 2021), etc; and (2) feed extracted features into a classifier for obtaining a pre-defined sentimental label.

In recent years, in-context learning (ICL) has achieved great success and changed the paradigm of NLP tasks. Many works adapt ICL to the sentiment analysis task: Qin et al. (2023b); Sun et al. (2023a) propose a series of strategies to improve ChatGPT's performance on the sentiment analysis task; Fei et al. (2023) propose a three-hop reasoning framework, which induces the implicit aspect, opinion, and finally the sentiment polarity for the implicit sentiment analysis task;*f* Zhang et al. (2023d) find that LLMs can achieve satisfactory performance on the binary sentiment classification task, but they underperform to the supervised baseline on more complex tasks (e.g., fine-grained sentiment analysis) that require deeper understanding or structured sentiment information.

2.2 Large Language Models and In-context Learning

Large language models (LLMs) (Wang et al., 2022a; Zhang et al., 2023b) are models trained on massive unlabeled text corpora with self-supervised learning techniques. Based on the model architecture, LLMs can be categorized into three types: (1) encoder-only models, which contain a text encoder and generate the input representations, such as BERT (Devlin et al., 2018)

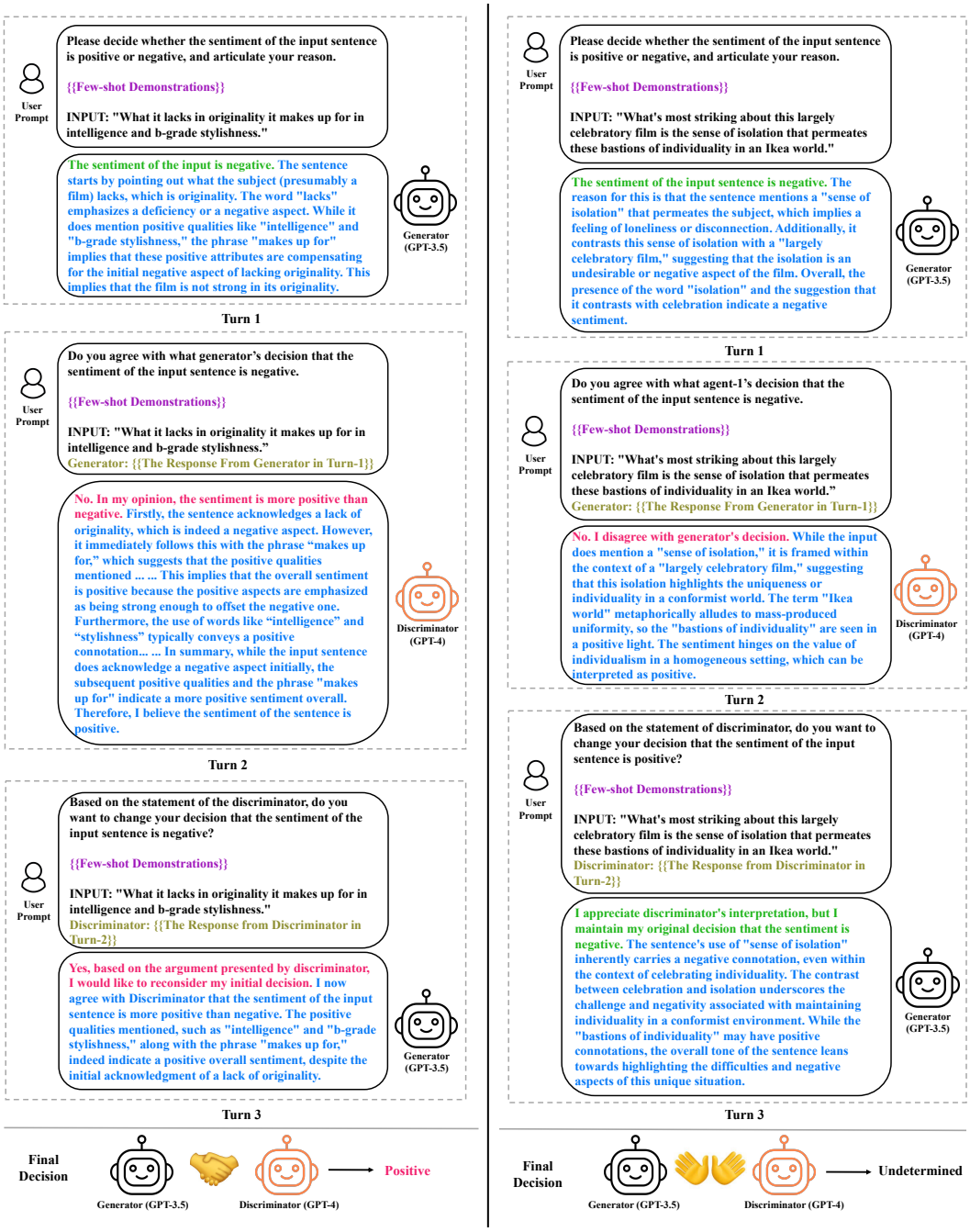

Figure 2: Illustration of the negotiation procedure. The left demonstration shows a case where an agreement on the positive sentiment is reached after turns turns, while the right demonstration shows a case where two LLMs fail to reach an agreement in three turns. Specifically, a user prompt includes four elements: a task description, few-shot demonstrations (abbreviate for short), an input, and a response from the last turn (if applicable). Responses from the generator or discriminator start with statements that the input contains either positive or negative sentiment, followed by rationales.

and its variants (Lan et al., 2019; Liu et al., 2019; Sun et al., 2020; Clark et al., 2020; Feng et al., 2020; Joshi et al., 2020; Sun et al., 2020, 2021); (2) decoder-only models, which have a decoder and generate text conditioned on the input text like GPT-series models (Radford et al., 2019; Brown et al., 2020; Keskar et al., 2019; Radford et al., 2019; Chowdhery et al., 2022; Ouyang et al., 2022; Zhang et al., 2022a; Scao et al., 2022; Zeng et al., 2022b; Touvron et al., 2023a; Peng et al., 2023; OpenAI, 2023); and (3) encoder-decoder models, which have a pair of encoder-decoder and generate text conditioned on the input representation, such as T5 (Raffel et al., 2020) and its variants (Lewis et al., 2019; Xue et al., 2020).

Starting with GPT-3 (Brown et al., 2020), LLMs have shown emerging capabilities (Wei et al., 2022a) and completed NLP tasks through in-context learning (ICL), where LLMs generate label-intensive text conditioned on a few annotated

examples without gradient updates. Many studies in the literature propose strategies for improving ICL performances on NLP tasks. Li and Liang (2021); Chevalier et al. (2023); Mu et al. (2023) optimize prompts in the continuous space. Liu et al. (2021a); Wan et al. (2023); Zhang et al. (2023a) search through the train set to retrieve k nearest neighbors of a test input as demonstrations. Zhang et al. (2022b); Sun et al. (2023b); Yao et al. (2023) decompose a task into a few sub-tasks and solve them step-by-step towards the final answer conditioned on LLM-generated reasoning chains. Sun et al. (2023a); Wang et al. (2023) propose to verify LLMs' results by conducting a new round of prompting; Liu et al. (2021b); Feng et al. (2023) use LLMs to generate natural language knowledge statements and integrate external knowledge statements into prompts.

2.3 The LLM collaboration

The LLM collaboration involves multiple LLMs working together to solve a given task. Specifically, the task is decomposed to several intermediate tasks, and each LLM is assigned to complete one intermediate task independently. The given task is solved after integrating or summarizing these intermediate results. The LLM collaboration approach can exploit the capabilities of LLMs, improve performances on complex tasks and enable to build complicated systems. Shinn et al. (2023); Sun et al. (2023a); Gero et al. (2023); Wang and Li (2023); Chen et al. (2023b) construct auxiliary tasks (e.g., reflection, verification tasks) and revise the response to the original task referring to the result of the auxiliary task. Talebirad and Nadiri (2023); Hong et al. (2023); Qian et al. (2023) assign characterize profiles (e.g., project manager, software engineer) to LLMs and gain performance boosts on character-specific tasks through behavior animations. Li et al. (2022); Zeng et al. (2022a); Chen et al. (2023a); Du et al. (2023); Liang et al. (2023) use a debate strategy in which multiple different LLMs propose their own responses to the given task and debate over multiple turns until getting a common final answer. Besides, Shen et al. (2023); Gao et al. (2023); Ge et al. (2023); Zhang et al. (2023c); Hao et al. (2023) employ one LLM as the task controller, which devises a plan for the given task, selects expert models for implementation and summarizes the responses of intermediate planned tasks. Other LLMs serve as task executors, completing intermediate tasks in their areas of expertise.

3 LLM Negotiation for Sentiment Analysis

3.1 Overview

In this section, we detail the multi-LLM negotiation framework for sentiment analysis: Two LLMs perform as the answer generator and discriminator. We refer to the interaction between the generator and the discriminator as a negotiation. The negotiation will repeat until a consensus is reached or the maximum number of negotiation turns is exceeded. Illustrations are shown in Figures 1 and 2.

3.2 Reasoning-infused generator

The generator is backboned by a large language model. We ask the answer generator based on the ICL paradigm through prompts, aiming to generate a step-by-step reasoning chain and a decision towards the sentiment polarity of the test input.

Prompts are composed of three elements: a task description, demonstrations, and a test input. The task description is a description of the task in natural language (e.g., "Please determine the overall sentiment of test input."); the test input is the textual input in the test set (e.g., "The sky is blue."); demonstrations are from the train set of the task. Each consists of three elements: input, reasoning chains, and sentimental decision.

For each test input, we first retrieve K nearest neighbors (input, sentiment decision) from the train set as demonstrations. Then, we transform demonstrations to (input, reasoning process, sentiment decision) triplets by prompting the generator to produce a reasoning chain. After concatenating the task description, demonstrations, and the test input, we forward the prompt to the generator, which will respond with a step-by-step reasoning chain and a sentimental decision.

3.3 Explanation-deriving discriminator

The discriminator is backboned by another LLM. After finishing the answer generating process, the answer discriminator is used to judge whether the decision made by the generator is correct and provide a reasonable explanation.

To accomplish this goal, we first construct prompts for the answer discriminator. The prompt is composed of four elements: a task description, demonstrations, a test input, and the response

from the answer generator. The task description is a piece of text that describes the task in natural language (e.g., "Please determine whether the decision is correct."). Each demonstration is composed of six elements: (input text, a reasoning chain, sentiment decision, discriminator attitude, discriminator explanations, discriminator decision) and constructed by prompting the answer discriminator to provide explanations of why the sentiment decision is correct for the input text.

Then we ask the discriminator with the construct prompt. The answer discriminator will respond with a text string, containing an attitude (i.e., yes, no) that denotes whether the discriminator agrees with the generator, explanations that explain why the discriminator agrees/disagrees with the generator, and a discriminator decision that determines the sentiment of the test input.

Why Two LLMs but Not One? There are two reasons for using two different LLMs separately for the generator and the discriminator rather than using a single LLM to act as two roles: (1) If an LLM makes a mistake as a generator due to incorrect reasoning, it is more likely that it will also make the same mistake as the discriminator as since generator and the discriminator from the same model are very likely to make similar rationales; (2) by using two separate models, we are able to take the advantage of the complementary abilities of the two models.

3.4 Role-flipped Negotiation

After two LLMs end with a negotiation, we ask them flip roles and initiate a new negotiation, where the second LLM acts as the generator, and the first LLM acts as the discriminator. We refer the interaction of two LLMs with flipped roles as role-flipped negotiation. Likewise, the role-flipped negotiation is ended until a consensus is reached or the maximum number of negotiation turns is exceeded.

When both negotiations result in an agreement and their decisions are the same, we can choose either decision as the final one since they are the same. If one of the negotiations fails to reach a consensus while the other reaches a decision, we choose the decision from the negotiation that reached a consensus as the final decision. However, if both negotiations reach a consensus but their decisions do not align, we will require the assistance of an additional Language Model (LLM), as will be explained in more detail below."

Introducing a third LLM If the decision from the two negotiations do not align, we introduce a third LLM and conduct the negotiation and role-flipped negotiation with each of the two aforementioned LLMs. Subsequently, we will get 6 negotiation results and vote on these results: the decision that appears most frequently is taken as the sentiment polarity of the input test.

4 Experiments

To evaluate the effectiveness of the proposed method, we use GPT-3.5, GPT-4 (OpenAI, 2023) and InstructGPT3.5 (Ouyang et al., 2022)[1] as backbones for the multi-model negotiation method. In this process, we use the fine-tuned RoBERTa-Large (Liu et al., 2019) as the similarity function for retrieving k nearest neighbors as demonstrations.

In the empirical study, we investigate the following three distinct ICL approaches, offering insights of integrating such methods for sentiment analysis.

- **Vanilla ICL**: the sentiment analysis task is finished by asking a LLM with a prompt to generate sentiment-intensive text without gradient updates. In practice, we conduct two sets of experiments under this setting with GPT3.5 and GPT-4, respectively.
- **Self-Negotiation**: the task is finished by using one LLM to discriminate and correct the answer generated by itself. We conduct two experiments with GPT3.5 and GPT-4 and get two results.
- **Negotiation with two LLMs**: the task is completed by employing two different LLMs to take turns performing as the answer generator and discriminator. Specifically, we conduct one set of experiment with GPT3.5 and GPT-4.

4.1 Datasets

We conduct experiments on six sentiment analysis datasets, including SST-2 (Socher et al., 2013), Movie Review (Zhang et al., 2015), Twitter (Rosenthal et al., 2019), Yelp-Binary (Zhang et al., 2015), Amazon-Binary (Zhang et al., 2015), and IMDB (Maas

[1] `text-davinci-003`

et al., 2011b). More details of the datasets are shown as follows:

- **SST-2** (Socher et al., 2013): SST-2 is a binary (i.e., positive, negative) sentiment classification dataset and contains movie review snippets from the Rotton Tomato. We follow Socher et al. (2013) and use the train, valid, test splits with the number of examples of 67,349, 872, 1,821, respectively.
- **Movie Review (MR)** (Zhang et al., 2015): Movie Reviews is a dataset for use in sentiment-analysis experiments. Available are collections of movie-review documents labeled with respect to their overall sentiment polarity (i.e., positive or negative).
- **Twitter** (Rosenthal et al., 2019): Twitter is a three-class (i.e., positive, negative, neutral) sentiment analysis dataset, aiming to detecting whether a piece of text expresses a sentiment polarity in respect to a specific topic, such as a person, a product, or an event. The dataset is origin a shared task at SemEval 2017, containing 50,333 examples in the train set and 12,284 examples in the test set.
- **Yelp-Binary** (Zhang et al., 2015): Yelp is a binary (i.e., positive, negative) sentiment analysis dataset, containing product reviews from Yelp. The dataset has 560,000 trainig samples and 38,000 testing samples.
- **Amazon-Binary** (Zhang et al., 2015): Amazon is a binary sentiment classification task, containing product reviews from Amazon with 3,600,000 examples in the train set and 400,000 examples in the test set.
- **IMDB** (Maas et al., 2011b): The IMDB dataset contains movie reviews along with their associated binary sentiment polarity labels. The dataset contains 50,000 reviews split evenly into 25k train and 25k test sets. The overall distribution of labels is balanced (25k positive and 25k negative).

We use accuracy as the evaluation metric.

4.2 Baselines

We use supervised neural network models and ICL approaches with LLMs as baselines for comparisons. For supervised methods, we choose the following four models:

- **DRNN** (Wang, 2018): incorporates position-invariance into RNN and CNN models by limiting the distance of information flow in neural networks.
- **RoBERTa** (Liu et al., 2019): is a reimplementation of BERT (Devlin et al., 2018) aiming to improve performances on NLP downstream tasks. In this paper, we report results achieved by fine-tuned RoBERTa-Large.
- **XLNet** (Yang et al., 2019): is a pre-trained autoregressive LM that integrates Transformer-XL (Dai et al., 2019) and enables to learn bidirectional contexts by maximizing the expected likelihood over all permutations of the factorization order.
- **UDA** (Xie et al., 2020): is short for Unsupervised Data Augmentation, which is a data augmentation strategy that employs a consistency loss function for unsupervised and supervised training stages. Performances in Table 1 are obtained by BERT-Large with UDA.
- **BERTweet** (Nguyen et al., 2020): is a pre-trained language model for English Tweets. The BERTweet has the same number of parameters as RoBERTa-Base.
- **EFL** (Wang et al., 2021): is backboned by RoBERTa-Large and fine-tuned on natural language entailment examples.

For ICL approaches, we report experimental results with LLMs from the following studies:

- Zhang et al. (2023d): presents a comprehensive study for applying LLMs (i.e., FLan-UL2, T5 and ChatGPT) on sentiment analysis tasks. Experimental results in the Table 1 are obtained in few-shot($k = 5$) settings.
- **InstructGPT-3.5** (Ouyang et al., 2022): is a large language model trained to follow human instructions. Experimental results in the Table 1 are achieved by the `text-davinci-003` model.
- **IDS** (Qin et al., 2023a): propose an Iterative Demonstration Selection (IDS) strategy to select demonstrations from diversity, similarity, and task-specific perspectives. Results shown in Table 1 are obtained by using GPT-3.5 (gpt-3.5-turbo).
- **GPT-4** (OpenAI, 2023): is a large multimodal model, achieving human-level performance on various NLP benchmarks.

	SST-2	Movie Review	Twitter	Yelp-Binary	Amazon-Binary	IMDB	**Average**
			Supervised Methods				
DRNN (Wang, 2018)	-	90.4	-	97.3	96.4	95.3	-
RoBERTa (Liu et al., 2019)	96.0	91.2	71.4	98.6	96.0	95.9	91.5
XLNet (Yang et al., 2019)	**97.0§**	-	-	**98.6§**	**97.9§**	**96.2§**	-
UDA (Xie et al., 2020)	-	-	-	97.9	96.5	95.8	-
BERTweet (Nguyen et al., 2020)	-	-	**71.6§**	-	-	-	-
EFL (Wang et al., 2021)	96.9	**92.5§**	-	-	-	96.1	-
			LLM ICL Baselines				
InstructGPT3.5 (Ouyang et al., 2022)	92.4	89.6	-	-	-	90.7	-
Zhang et al. (2023d)							
- w/ Flan-UL2	97.4	93.8	47.9	-	-	-	-
- w/ T5	91.4	85.7	53.2	92.4	-	90.0	-
- w/ GPT-3.5	95.3	90.2	64.3	-	-	-	-
IDS (Qin et al., 2023a)	95.8	-	-	94.2	95.7	-	-
GPT-4 (OpenAI, 2023)	92.5	-	-	94.2	-	-	-
			Our Implementation				
Vanilla ICL							
- w/ GPT-3.5	92.7	90.2	65.2	93.8	84.8	90.6	86.2
- w/ GPT-4	93.2	89.4	69.5	95.2	83.5	88.5	86.6
Self-Negotiation							
- w/ GPT-3.5	93.2	90.6	66.8	94.5	86.0	91.7	87.1
- w/ GPT-4	93.3	90.3	72.2	95.5	84.3	89.7	87.6
Negotiation with LLMs							
- w/ GPT-3.5+GPT-4	93.8	92.3	74.3	96.3	86.9	94.0	89.6
- w/ GPT-3.5+GPT-4+InstructGPT3.5	94.1	92.7	74.6	96.3	87.2	94.5	89.8

Table 1: Accuracy performances of different settings on benchmarks. Performances with § denote current state-of-the-art.

- **Self-negotiation**: The same LLM acts as both the roles of the generator and the discriminator.

4.3 Results and analysis

Experiment results are shown in Table 1. As can be seen in the table, compared to vanilla ICL, following the generate-discriminate paradigm with one LLM (self-negotiation) receives performance gains on six sentiment analysis datasets: GPT-3.5 gains +0.9 on average; GPT-4 receives +1.0 acc on average. This phenomenon illustrates that the LLM, performing as the answer discriminator, can correct a portion of errors caused by the task generator.

We also observe that using two different LLMs as the task generator and task discriminator in turn introduces significant performance improvements compared to merely using one model. Negotiations with two LLMs outperform the self-negotiation method by +1.7, +2.1, and +2.3 in terms of accuracy on MR, Twitter, IMDB datasets, respectively. The reason for this phenomenon is that using two different LLMs finish the sentiment analysis task through negotiations can take the advantage of different understandings of the given input and unleash the power of two LLMs, leading to more accurate decisions.

We also find that when introduce a third LLM to resolve the disagreement between the flippled-roled negotiations, additional performance boost can be obtained. This demonstrates that the third LLM can resolve conflicts between two LLMs through multiple negotiations and improve performances on the sentiment analysis task. It is noteworthy that the multi-model negotiation method outperforms the supervised method RoBERTa-Large by +0.9 on the MR dataset, and bridges the gap between vanilla ICL and the supervised method: achieving 94.1 (+1.4) accuracy on SST-2; 92.1 (+2.7) on Twitter; 96.3 (+2.5) on Yelp-2; 87.2 (+3.7) on Amazon-2; and 94.5 (+6.0) on IMDB dataset.

5 Ablation Studies

In this section, we perform ablation studies on the Twitter dataset to better understand the mechanism behind the negotiation framework.

5.1 Who takes which role matters

In the negotiation framework, there are two roles, the generator and the discriminator, which two separate LLMs take. Table 2 shows the performance for setups where GPT-3.5 and GPT-4 take different roles.

As can be seen, when GPT-3.5 acts as the

G	D	ACC
GPT-3.5	-	65.2
GPT-4	-	69.5
GPT-3.5	GPT-3.5	66.8
GPT-3.5	GPT-4	65.2
GPT-4	GPT-3.5	72.8
GPT-4	GPT-4	72.2

Table 2: Performance on the Twitter dataset with GPT-3.5 and GPT-4 taking different roles. G denotes generator and D denotes discriminator.

	G3.5-D4	G4-D3.5
2 turns agree	65%	76%
3 turns agree	29%	21%
3 turns disagree	6%	3%

Table 3: Consensus percentage for different setups on the Twitter dataset. G3.5-D4 denotes GPT-3.5 acts as the generator and GPT-4 acts as the discriminator.

generator, and GPT-4 acts as the discriminator (G3.5-D4 for short), the performance (68.8) is better than single GPT-3.5 without negotiation (65.2), but worse than single GPT-4 without negotiation (69.5). In contrast, negotiation-based configurations with GPT-4 acting as the generator (G4-D3.5 and G4-D4) consistently outperforms standalone GPT-4 or GPT-3.5 models without negotiation. These results underscore the pivotal role that the generator plays in influencing the negotiation outcome. Furthermore, we observe G4-D3.5 can beat G4-D4. We attribute such advantage to the hypothesis that utilizing heterogeneous LLMs for distinct roles could optimize the negotiation's performance.

5.2 Consensus Percentage

Table 3 consensus percentage for different setups. As can be seen, when GPT-4 acts as the generator, the negotiation is more likely to reach a consensus, or reach a consensus in fewer turns. The explanation is intuitive: for the twitter task, we can see from table 1 that GPT-4 obtains better performances that GPT-3.5, which means the reasoning process for GPT-4 is more sensible than 3.5, making the decision of the former more likely to be agreed on.

5.3 Effect of the Reasoning Process

In the negotiation process, LLMs are asked to articulate the reason process, a strategy akin to CoT(Wei et al., 2022b). We examine the importance for listing reasons in negotiation by removing the reasoning process and asking LLMs to only output decisions. Results are shown in Table 4. As can be seen, for the three setups, single GPT-3.5, where only GPT-3.5 is used without negotiation, single GPT-4, where only GPT-4 is used without negotiation, and GPT-3.5+GPT-4 where negotiation is employed, performances all degrade when the reasoning process is removed. But interestingly, we see a greater degrade (-2.3) for the negotiation than the single model setup (-1.2 for single-GPT-3.5 and -0.9 for single-GPT-4). This is in accord with our expectation as the reasoning process is of greater significance in the negotiation setup.

Model	Reason	ACC
single GPT-3.5	w	65.2
single GPT-3.5	wo	64.0 (-1.2)
single GPT-4	w	69.5
single GPT-4	wo	68.6 (-0.9)
GPT-3.5+GPT-4	w	74.6
GPT-3.5+GPT-4	wo	72.3 (-2.3)

Table 4: Effect of removing the reasoning process on the Twitter dataset.

6 Conclusion

In this paper, we investigate the limitations of singular LLM-based sentiment analysis methods and introduce a novel role-flipping multi-LLM negotiation method to enhance both the accuracy and interpretability of sentiment categorizations. Empirical findings on multiple benchmarks show the superiority of our approach compared to traditional ICL and many supervised methods. Future work could explore optimizing the framework for speed and resource consumption, adapting the underlying principles to other NLP tasks, and designing explicit negotiation modules that identify and mitigate the impact of biases and decoding errors present in individual LLMs.

Limitations

This paper acknowledges several inherent limitations associated with the use of large language models (LLMs), particularly in the context of negations. Firstly, LLMs can sometimes struggle with accurately interpreting and responding to negations, leading to responses

that may not fully align with the intended meaning. This limitation stems from the complex nature of negation in human language, which often requires a deep understanding of context, nuance, and implicit knowledge.

References

Rohan Anil, Andrew M Dai, Orhan Firat, Melvin Johnson, Dmitry Lepikhin, Alexandre Passos, Siamak Shakeri, Emanuel Taropa, Paige Bailey, Zhifeng Chen, et al. 2023. Palm 2 technical report. *arXiv preprint arXiv:2305.10403*.

Stefano Baccianella, Andrea Esuli, and Fabrizio Sebastiani. 2010. Sentiwordnet 3.0: An enhanced lexical resource for sentiment analysis and opinion mining. In *International Conference on Language Resources and Evaluation*.

Jinze Bai, Shuai Bai, Yunfei Chu, Zeyu Cui, Kai Dang, Xiaodong Deng, Yang Fan, Wenbin Ge, Yu Han, Fei Huang, et al. 2023. Qwen technical report. *arXiv preprint arXiv:2309.16609*.

Rushlene Kaur Bakshi, Navneet Kaur, Ravneet Kaur, and Gurpreet Kaur. 2016. Opinion mining and sentiment analysis. *2016 3rd International Conference on Computing for Sustainable Global Development (INDIACom)*, pages 452–455.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Justin Chih-Yao Chen, Swarnadeep Saha, and Mohit Bansal. 2023a. Reconcile: Round-table conference improves reasoning via consensus among diverse llms. *arXiv preprint arXiv:2309.13007*.

Xinyun Chen, Maxwell Lin, Nathanael Schärli, and Denny Zhou. 2023b. Teaching large language models to self-debug. *ArXiv*, abs/2304.05128.

Alexis Chevalier, Alexander Wettig, Anirudh Ajith, and Danqi Chen. 2023. Adapting language models to compress contexts. *arXiv preprint 2305.14788*.

Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, et al. 2022. Palm: Scaling language modeling with pathways. *arXiv preprint arXiv:2204.02311*.

Kevin Clark, Minh-Thang Luong, Quoc V Le, and Christopher D Manning. 2020. Electra: Pre-training text encoders as discriminators rather than generators. *arXiv preprint arXiv:2003.10555*.

Junqi Dai, Hang Yan, Tianxiang Sun, Pengfei Liu, and Xipeng Qiu. 2021. Does syntax matter? a strong baseline for aspect-based sentiment analysis with roberta. In *North American Chapter of the Association for Computational Linguistics*.

Z Dai, Z Yang, Y Yang, J Carbonell, Q Le, and R Transformer-XL Salakhutdinov. 2019. Attentive language models beyond a fixed-length context. 2019. *arXiv preprint arXiv:1901.02860*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

Yilun Du, Shuang Li, Antonio Torralba, Joshua B Tenenbaum, and Igor Mordatch. 2023. Improving factuality and reasoning in language models through multiagent debate. *arXiv preprint arXiv:2305.14325*.

Hao Fei, Bobo Li, Qian Liu, Lidong Bing, Fei Li, and Tat-Seng Chua. 2023. Reasoning implicit sentiment with chain-of-thought prompting. *arXiv preprint arXiv:2305.11255*.

Shangbin Feng, Vidhisha Balachandran, Yuyang Bai, and Yulia Tsvetkov. 2023. Factkb: Generalizable factuality evaluation using language models enhanced with factual knowledge. *ArXiv*, abs/2305.08281.

Zhangyin Feng, Daya Guo, Duyu Tang, Nan Duan, Xiaocheng Feng, Ming Gong, Linjun Shou, Bing Qin, Ting Liu, Daxin Jiang, et al. 2020. Codebert: A pre-trained model for programming and natural languages. *arXiv preprint arXiv:2002.08155*.

Difei Gao, Lei Ji, Luowei Zhou, Kevin Qinghong Lin, Joya Chen, Zihan Fan, and Mike Zheng Shou. 2023. Assistgpt: A general multi-modal assistant that can plan, execute, inspect, and learn. *arXiv preprint arXiv:2306.08640*.

Yingqiang Ge, Wenyue Hua, Jianchao Ji, Juntao Tan, Shuyuan Xu, and Yongfeng Zhang. 2023. Openagi: When llm meets domain experts. *arXiv preprint arXiv:2304.04370*.

Zelalem Gero, Chandan Singh, Hao Cheng, Tristan Naumann, Michel Galley, Jianfeng Gao, and Hoifung Poon. 2023. Self-verification improves few-shot clinical information extraction. *arXiv preprint arXiv:2306.00024*.

Alec Go, Richa Bhayani, and Lei Huang. 2009. Twitter sentiment classification using distant supervision. *CS224N project report, Stanford*, 1(12):2009.

Ziyu Guan, Long Chen, Wei Zhao, Yi Zheng, Shulong Tan, and Deng Cai. 2016. Weakly-supervised deep learning for customer review sentiment classification. In *International Joint Conference on Artificial Intelligence*.

Chinnappa Guggilla, Tristan Miller, and Iryna Gurevych. 2016. Cnn- and lstm-based claim classification in online user comments. In *International Conference on Computational Linguistics*.

Rui Hao, Linmei Hu, Weijian Qi, Qingliu Wu, Yirui Zhang, and Liqiang Nie. 2023. Chatllm network: More brains, more intelligence. *arXiv preprint arXiv:2304.12998*.

Sirui Hong, Xiawu Zheng, Jonathan Chen, Yuheng Cheng, Ceyao Zhang, Zili Wang, Steven Ka Shing Yau, Zijuan Lin, Liyang Zhou, Chenyu Ran, et al. 2023. Metagpt: Meta programming for multi-agent collaborative framework. *arXiv preprint arXiv:2308.00352*.

Mandar Joshi, Danqi Chen, Yinhan Liu, Daniel S Weld, Luke Zettlemoyer, and Omer Levy. 2020. Spanbert: Improving pre-training by representing and predicting spans. *Transactions of the Association for Computational Linguistics*, 8:64–77.

Nal Kalchbrenner, Edward Grefenstette, and Phil Blunsom. 2014. A convolutional neural network for modelling sentences. *arXiv preprint arXiv:1404.2188*.

Nitish Shirish Keskar, Bryan McCann, Lav R Varshney, Caiming Xiong, and Richard Socher. 2019. Ctrl: A conditional transformer language model for controllable generation. *arXiv preprint arXiv:1909.05858*.

Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. 2022. Large language models are zero-shot reasoners. *Advances in neural information processing systems*, 35:22199–22213.

Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. 2019. Albert: A lite bert for self-supervised learning of language representations. *arXiv preprint arXiv:1909.11942*.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Ves Stoyanov, and Luke Zettlemoyer. 2019. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *arXiv preprint arXiv:1910.13461*.

Rui Li, Guoyin Wang, and Jiwei Li. 2023. Are human-generated demonstrations necessary for in-context learning? *arXiv preprint arXiv:2309.14681*.

Shuang Li, Yilun Du, Joshua B Tenenbaum, Antonio Torralba, and Igor Mordatch. 2022. Composing ensembles of pre-trained models via iterative consensus. *arXiv preprint arXiv:2210.11522*.

Xiang Lisa Li and Percy Liang. 2021. Prefix-tuning: Optimizing continuous prompts for generation. *arXiv preprint arXiv:2101.00190*.

Tian Liang, Zhiwei He, Wenxiang Jiao, Xing Wang, Yan Wang, Rui Wang, Yujiu Yang, Zhaopeng Tu, and Shuming Shi. 2023. Encouraging divergent thinking in large language models through multi-agent debate. *ArXiv*, abs/2305.19118.

Hunter Lightman, Vineet Kosaraju, Yura Burda, Harri Edwards, Bowen Baker, Teddy Lee, Jan Leike, John Schulman, Ilya Sutskever, and Karl Cobbe. 2023. Let's verify step by step. *arXiv preprint arXiv:2305.20050*.

Yuxiao Lin, Yuxian Meng, Xiaofei Sun, Qinghong Han, Kun Kuang, Jiwei Li, and Fei Wu. 2021. Bertgcn: Transductive text classification by combining gcn and bert. *arXiv preprint arXiv:2105.05727*.

Jiachang Liu, Dinghan Shen, Yizhe Zhang, Bill Dolan, Lawrence Carin, and Weizhu Chen. 2021a. What makes good in-context examples for gpt-3? *arXiv preprint arXiv:2101.06804*.

Jiacheng Liu, Alisa Liu, Ximing Lu, Sean Welleck, Peter West, Ronan Le Bras, Yejin Choi, and Hannaneh Hajishirzi. 2021b. Generated knowledge prompting for commonsense reasoning. *arXiv preprint arXiv:2110.08387*.

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. *arXiv preprint arXiv:1907.11692*.

Yao Lu, Max Bartolo, Alastair Moore, Sebastian Riedel, and Pontus Stenetorp. 2021. Fantastically ordered prompts and where to find them: Overcoming few-shot prompt order sensitivity. *arXiv preprint arXiv:2104.08786*.

Andrew Maas, Raymond E Daly, Peter T Pham, Dan Huang, Andrew Y Ng, and Christopher Potts. 2011a. Learning word vectors for sentiment analysis. In *Proceedings of the 49th annual meeting of the association for computational linguistics: Human language technologies*, pages 142–150.

Andrew L. Maas, Raymond E. Daly, Peter T. Pham, Dan Huang, A. Ng, and Christopher Potts. 2011b. Learning word vectors for sentiment analysis. In *Annual Meeting of the Association for Computational Linguistics*.

Walaa Medhat, Ahmed Hussein Hassan, and Hoda Korashy. 2014. Sentiment analysis algorithms and applications: A survey. *Ain Shams Engineering Journal*, 5:1093–1113.

Abhijit Mishra, Kuntal Dey, and Pushpak Bhattacharyya. 2017. Learning cognitive features from gaze data for sentiment and sarcasm classification using convolutional neural network. In *Annual Meeting of the Association for Computational Linguistics*.

Jesse Mu, Xiang Lisa Li, and Noah Goodman. 2023. Learning to compress prompts with gist tokens. *arXiv preprint arXiv:2304.08467*.

Dat Quoc Nguyen, Thanh Vu, and Anh Tuan Nguyen. 2020. Bertweet: A pre-trained language model for english tweets. *arXiv preprint arXiv:2005.10200*.

OpenAI. 2023. Gpt-4 technical report. *ArXiv*, abs/2303.08774.

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. 2022. Training language models to follow instructions with human feedback. *arXiv preprint arXiv:2203.02155*.

Bo Pang and Lillian Lee. 2008. Opinion mining and sentiment analysis. *Found. Trends Inf. Retr.*, 2:1–135.

Lotem Peled and Roi Reichart. 2017. Sarcasm sign: Interpreting sarcasm with sentiment based monolingual machine translation. *arXiv preprint arXiv:1704.06836*.

Baolin Peng, Chunyuan Li, Pengcheng He, Michel Galley, and Jianfeng Gao. 2023. Instruction tuning with gpt-4. *arXiv preprint arXiv:2304.03277*.

Minh Hieu Phan and Philip Ogunbona. 2020. Modelling context and syntactical features for aspect-based sentiment analysis. In *Annual Meeting of the Association for Computational Linguistics*.

Chen Qian, Xin Cong, Cheng Yang, Weize Chen, Yusheng Su, Juyuan Xu, Zhiyuan Liu, and Maosong Sun. 2023. Communicative agents for software development. *arXiv preprint arXiv:2307.07924*.

Qiao Qian, Minlie Huang, Jinhao Lei, and Xiaoyan Zhu. 2016. Linguistically regularized lstms for sentiment classification. *arXiv preprint arXiv:1611.03949*.

Chengwei Qin, Aston Zhang, Anirudh Dagar, and Wenming Ye. 2023a. In-context learning with iterative demonstration selection.

Chengwei Qin, Aston Zhang, Zhuosheng Zhang, Jiaao Chen, Michihiro Yasunaga, and Diyi Yang. 2023b. Is chatgpt a general-purpose natural language processing task solver? *arXiv preprint arXiv:2302.06476*.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog*.

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. *The Journal of Machine Learning Research*, 21(1):5485–5551.

Sara Rosenthal, Noura Farra, and Preslav Nakov. 2019. Semeval-2017 task 4: Sentiment analysis in twitter. *arXiv preprint arXiv:1912.00741*.

Teven Le Scao, Angela Fan, Christopher Akiki, Ellie Pavlick, Suzana Ilić, Daniel Hesslow, Roman Castagné, Alexandra Sasha Luccioni, François Yvon, Matthias Gallé, et al. 2022. Bloom: A 176b-parameter open-access multilingual language model. *arXiv preprint arXiv:2211.05100*.

Timo Schick, Jane Dwivedi-Yu, Roberto Dessì, Roberta Raileanu, Maria Lomeli, Luke Zettlemoyer, Nicola Cancedda, and Thomas Scialom. 2023. Toolformer: Language models can teach themselves to use tools. *arXiv preprint arXiv:2302.04761*.

Yongliang Shen, Kaitao Song, Xu Tan, Dongsheng Li, Weiming Lu, and Yueting Zhuang. 2023. Hugginggpt: Solving ai tasks with chatgpt and its friends in huggingface. *arXiv preprint arXiv:2303.17580*.

Noah Shinn, Beck Labash, and Ashwin Gopinath. 2023. Reflexion: an autonomous agent with dynamic memory and self-reflection. *ArXiv*, abs/2303.11366.

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D Manning, Andrew Y Ng, and Christopher Potts. 2013. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 conference on empirical methods in natural language processing*, pages 1631–1642.

Xiaofei Sun, Linfeng Dong, Xiaoya Li, Zhen Wan, Shuhe Wang, Tianwei Zhang, Jiwei Li, Fei Cheng, Lingjuan Lyu, Fei Wu, et al. 2023a. Pushing the limits of chatgpt on nlp tasks. *arXiv preprint arXiv:2306.09719*.

Xiaofei Sun, Xiaoya Li, Jiwei Li, Fei Wu, Shangwei Guo, Tianwei Zhang, and Guoyin Wang. 2023b. Text classification via large language models. *arXiv preprint arXiv:2305.08377*.

Yu Sun, Shuohuan Wang, Yukun Li, Shikun Feng, Hao Tian, Hua Wu, and Haifeng Wang. 2020. Ernie 2.0: A continual pre-training framework for language understanding. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34.

Zijun Sun, Xiaoya Li, Xiaofei Sun, Yuxian Meng, Xiang Ao, Qing He, Fei Wu, and Jiwei Li. 2021. Chinesebert: Chinese pretraining enhanced by glyph and pinyin information. *arXiv preprint arXiv:2106.16038*.

Yashar Talebirad and Amirhossein Nadiri. 2023. Multi-agent collaboration: Harnessing the power of intelligent llm agents. *arXiv preprint arXiv:2306.03314*.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. 2023a. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023b. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Duy-Tin Vo and Yue Zhang. 2015. Target-dependent twitter sentiment classification with rich automatic features. In *Twenty-fourth international joint conference on artificial intelligence*.

Zhen Wan, Fei Cheng, Zhuoyuan Mao, Qianying Liu, Haiyue Song, Jiwei Li, and Sadao Kurohashi. 2023. Gpt-re: In-context learning for relation extraction using large language models. *arXiv preprint arXiv:2305.02105*.

Baoxin Wang. 2018. Disconnected recurrent neural networks for text categorization. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2311–2320.

Danqing Wang and Lei Li. 2023. Learn from mistakes through cooperative interaction with study assistant. *ArXiv*, abs/2305.13829.

Haifeng Wang, Jiwei Li, Hua Wu, Eduard Hovy, and Yu Sun. 2022a. Pre-trained language models and their applications. *Engineering*.

Jin Wang, Liang-Chih Yu, K Robert Lai, and Xuejie Zhang. 2016a. Dimensional sentiment analysis using a regional cnn-lstm model. In *Proceedings of the 54th annual meeting of the association for computational linguistics (volume 2: Short papers)*, pages 225–230.

Shuhe Wang, Xiaofei Sun, Xiaoya Li, Rongbin Ouyang, Fei Wu, Tianwei Zhang, Jiwei Li, and Guoyin Wang. 2023. Gpt-ner: Named entity recognition via large language models. *arXiv preprint arXiv:2304.10428*.

Sinong Wang, Han Fang, Madian Khabsa, Hanzi Mao, and Hao Ma. 2021. Entailment as few-shot learner. *arXiv preprint arXiv:2104.14690*.

Xingyou Wang, Weijie Jiang, and Zhiyong Luo. 2016b. Combination of convolutional and recurrent neural network for sentiment analysis of short texts. In *Proceedings of COLING 2016, the 26th international conference on computational linguistics: Technical papers*, pages 2428–2437.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. 2022b. Self-consistency improves chain of thought reasoning in language models. *arXiv preprint arXiv:2203.11171*.

Jason Wei, Yi Tay, Rishi Bommasani, Colin Raffel, Barret Zoph, Sebastian Borgeaud, Dani Yogatama, Maarten Bosma, Denny Zhou, Donald Metzler, et al. 2022a. Emergent abilities of large language models. *arXiv preprint arXiv:2206.07682*.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Ed Chi, Quoc Le, and Denny Zhou. 2022b. Chain of thought prompting elicits reasoning in large language models. *arXiv preprint arXiv:2201.11903*.

Qizhe Xie, Zihang Dai, Eduard Hovy, Thang Luong, and Quoc Le. 2020. Unsupervised data augmentation for consistency training. *Advances in neural information processing systems*, 33:6256–6268.

Linting Xue, Noah Constant, Adam Roberts, Mihir Kale, Rami Al-Rfou, Aditya Siddhant, Aditya Barua, and Colin Raffel. 2020. mt5: A massively multilingual pre-trained text-to-text transformer. *arXiv preprint arXiv:2010.11934*.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R Salakhutdinov, and Quoc V Le. 2019. Xlnet: Generalized autoregressive pretraining for language understanding. *Advances in neural information processing systems*, 32.

Shunyu Yao, Dian Yu, Jeffrey Zhao, Izhak Shafran, Thomas L Griffiths, Yuan Cao, and Karthik Narasimhan. 2023. Tree of thoughts: Deliberate problem solving with large language models. *arXiv preprint arXiv:2305.10601*.

Jianfei Yu and Jing Jiang. 2016. Learning sentence embeddings with auxiliary tasks for cross-domain sentiment classification. In *Conference on Empirical Methods in Natural Language Processing*.

Andy Zeng, Maria Attarian, Brian Ichter, Krzysztof Choromanski, Adrian Wong, Stefan Welker, Federico Tombari, Aveek Purohit, Michael Ryoo, Vikas Sindhwani, et al. 2022a. Socratic models: Composing zero-shot multimodal reasoning with language. *arXiv preprint arXiv:2204.00598*.

Aohan Zeng, Xiao Liu, Zhengxiao Du, Zihan Wang, Hanyu Lai, Ming Ding, Zhuoyi Yang, Yifan Xu, Wendi Zheng, Xiao Xia, et al. 2022b. Glm-130b: An open bilingual pre-trained model. *arXiv preprint arXiv:2210.02414*.

Lei Zhang and B. Liu. 2012. Sentiment analysis and opinion mining. In *Encyclopedia of Machine Learning and Data Mining*.

Lei Zhang, Shuai Wang, and B. Liu. 2018. Deep learning for sentiment analysis: A survey. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 8.

Mozhi Zhang, Hang Yan, Yaqian Zhou, and Xipeng Qiu. 2023a. Promptner: A prompting method for few-shot named entity recognition via k nearest neighbor search. *arXiv preprint arXiv:2305.12217*.

Shengyu Zhang, Linfeng Dong, Xiaoya Li, Sen Zhang, Xiaofei Sun, Shuhe Wang, Jiwei Li, Runyi Hu, Tianwei Zhang, Fei Wu, et al. 2023b. Instruction tuning for large language models: A survey. *arXiv preprint arXiv:2308.10792*.

Shujian Zhang, Chengyue Gong, Lemeng Wu, Xingchao Liu, and Mingyuan Zhou. 2023c. Automl-gpt: Automatic machine learning with gpt. *arXiv preprint arXiv:2305.02499*.

Susan Zhang, Stephen Roller, Naman Goyal, Mikel Artetxe, Moya Chen, Shuohui Chen, Christopher Dewan, Mona Diab, Xian Li, Xi Victoria Lin, et al. 2022a. Opt: Open pre-trained transformer language models. *arXiv preprint arXiv:2205.01068*.

Wenxuan Zhang, Yue Deng, Bing Liu, Sinno Jialin Pan, and Lidong Bing. 2023d. Sentiment analysis in the era of large language models: A reality check. *arXiv preprint arXiv:2305.15005*.

Xiang Zhang, Junbo Zhao, and Yann LeCun. 2015. Character-level convolutional networks for text classification. *Advances in neural information processing systems*, 28.

Zhuosheng Zhang, Aston Zhang, Mu Li, and Alex Smola. 2022b. Automatic chain of thought prompting in large language models. *arXiv preprint arXiv:2210.03493*.