Meta-Task Prompting Elicits Embedding from Large Language Models

Anonymous ACL submission

Abstract

In this work, we introduce a new unsupervised embedding method, Meta-Task Prompting with Explicit One-Word Limitation (MetaEOL), for generating high-quality sentence embeddings from Large Language Models (LLMs) without the need for model fine-tuning or task-specific engineering. Leveraging meta-task prompting, MetaEOL guides LLMs to produce embeddings through a series of carefully designed prompts that address multiple representational aspects. Our comprehensive experiments demonstrate that embeddings averaged from various meta-tasks yield competitive performance on Semantic Textual Similarity (STS) benchmarks and excel in downstream tasks, surpassing contrastive-trained models. Our findings suggest a new scaling law for embedding generation, offering a versatile, resource-efficient approach for embedding extraction across diverse sentence-centric scenarios.[1]

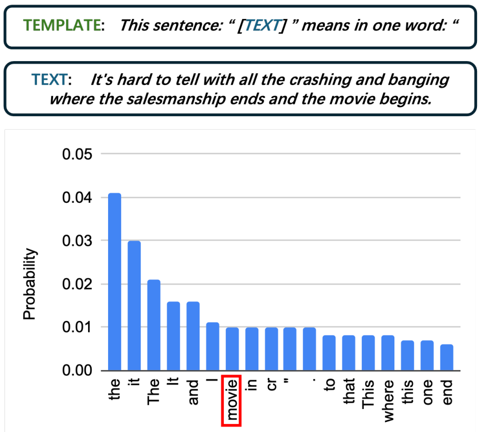

Figure 1: The highest decoding probabilities are largely allocated to stop words that carry little useful information when conducting a meaning compression prompting, even if employing a constraint of "*in one word*" following (Jiang et al., 2023b). Although the general semantic, movie, is contained, other aspects of this sentence are missing, like sentiments.

1 Introduction

The advent of Large Language Models (LLMs) such as GPT-3 (Brown et al., 2020) and LLaMA (Touvron et al., 2023a) has marked a significant milestone in the field of natural language processing (NLP), introducing promising unsupervised methods for various NLP tasks by leveraging task-related instructions or prompts. These tasks also include the generation of sentence embeddings, which aims to produce sentence representations that can be applied across a spectrum of scenarios, ranging from intrinsic tasks like Semantic Textual Similarity (STS) (Agirre et al., 2012a; Cer et al., 2017b) to downstream tasks including information retrieval (Mitra et al., 2017; Izacard et al., 2021), sentiment categorization (Ke et al., 2020), and beyond. By employing specific prompts (Jiang et al., 2023b, 2022a), researchers have begun to explore the potential of extracting meaningful sentence embeddings directly from the hidden states of LLMs without the need for explicit training on embedding-specific tasks. The significance of zero-resource settings – where embeddings are generated without any fine-tuning or in-context learning – cannot be overstated.

Initial efforts in this domain, as highlighted by works like (Jiang et al., 2023b, 2022a; Liu et al., 2023a), have focused on unsupervised techniques that extract sentence representations directly from LLMs. These methods typically involve using fill-in-the-blanks prompts, such as *This sentence: "[TEXT]" means in one word:"* (Jiang et al., 2023b), to embed a sentence into a single token representation, achieved by extracting output hidden states of the last token as the sentence's embedding. Despite promising, they also reveal the inherent challenges where embeddings may be overly simplistic or misaligned with the intended semantic nuances of the

[1] Our anonymous code link: `https://anonymous.4open.science/r/MetaEOL`.

sentences.

In our pilot experiment, we show that the previous prompt-based method (Jiang et al., 2023b) struggles to capture a comprehensive meaning for a sentence, especially when the usage of the sentence is associated with multiple aspects. As illustrated in Figure 1, when probing probability distribution for the next token during decoding, which reflects the embedding quality of the last token, the highest probabilities are mostly distributed to frequent stop words. Although the general semantic — movie — is contained, other meaningful aspects like sentiments are missing.

A straightforward solution to mitigate this issue is to demonstrate LLMs with task-specific instructions. This approach involves instructing the model with prompts explicitly designed for a particular task, thereby tailoring the embeddings to better suit the specific requirements of that task. However, considering the realm of NLP encompasses thousands of distinct tasks (Mishra et al., 2022; Wang et al., 2022; Chung et al., 2022), this would be impractical due to computational and storage challenges. Furthermore, task-specific embeddings, while effective for their designated tasks, often fail to generalize well across different tasks.

Drawing on the principles of the usage-based theory of language acquisition (Tomasello, 2009), which asserts that the essence of meaning is rooted in the practical utilization of language, we aim to introduce a novel, unsupervised approach for the extraction of high-quality sentence embeddings directly from LLMs. Our method does not confine itself to deriving embeddings from a singular, overarching meaning. Instead, it embraces the concept of meta-task prompting, inspired by the burgeoning fields of meta-task promoted training (Sanh et al., 2022) and hyper-prompt (He et al., 2022) techniques. By defining a suite of meta-tasks, each tailored to a distinct application context, MetaEOL prompts LLMs to consider multiple representational tokens from a variety of perspectives. This multifaceted approach enables the extraction of more diverse and nuanced contextualized token embeddings that collectively form a comprehensive sentence embedding.

Extensive experiments empirically show that: (I) Simply averaging embeddings from different meta-tasks without any training leads to general embeddings that are competitive to contrastive-trained models on STS tasks and can achieve the best average result in downstream tasks. (II) Incrementally integrating more meta-tasks (ranging from one to four) yields consistent improvements across STS tasks, showcasing high generalities, and highlighting the significant impact of meta-task integration on overall performance. (III) The final layer is not always the most effective for STS tasks and with a simple proportional layer selection strategy, we achieve the best results with a 70B model, which points to a potential scaling law.

2 Related Work

Sentence Embeddings. Sentence embeddings aim to encapsulate the semantic content of sentences into fixed-sized vector representations. Recent developments in contrastive learning have proven to be highly effective for generating sentence embeddings, under both unsupervised and supervised settings (Gao et al., 2021; Jiang et al., 2022a; Chuang et al., 2022; Wu et al., 2022). For instance, SimCSE (Gao et al., 2021) utilizes different dropout masks as a form of noise to create positive pairs in an unsupervised fashion, while in a supervised setting, models like Sentence-BERT (Reimers and Gurevych, 2019) leverage natural language inference (NLI) datasets to construct positive and negative pairs. Additionally, Su et al. (2023) and Asai et al. (2023) show that training with a large amount of tasks with annotated instructions can enable the model to generate embeddings tailored to different downstream tasks. In contrast, our approach MetaEOL demonstrates the potential of utilizing LLMs directly to generate instruction-followed embeddings without necessitating any training.

Large Language Models for Sentence Representation. Recent studies have explored the application of LLMs for enhancing sentence embeddings through data augmentation techniques (Cheng et al., 2023; Zhang et al., 2023). Notably, Sentence-T5 (Ni et al., 2022) employs contrastive learning on models with billions of parameters, demonstrating consistent performance improvements correlating with increases in model size. More recently, Liu et al. (2023b) represents sentences through the distribution of possible text continuations, comparing the distributional similarity between sentences. This method, although effective, necessitates the generation of 20 trajectories, each up to 20 tokens in length, making it computationally intensive. Moreover, this type of representation can not directly be used for downstream tasks.

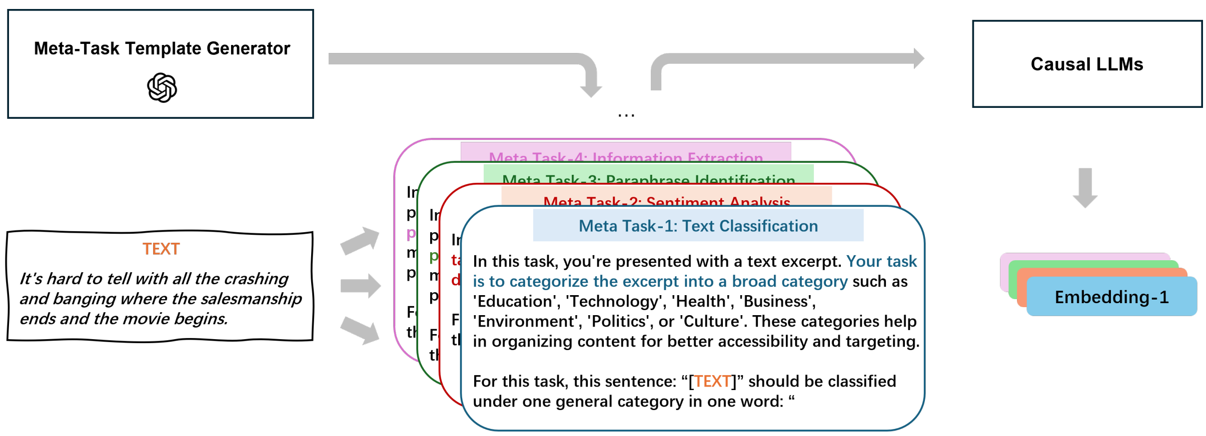

Figure 2: The workflow of our method (MetaEOL). We use the prompt in Appendix A.1 to prompt ChatGPT-4 to generate templates. Each input sentence will be decorated with multiple task-specific templates, indicating its various intended usage scenarios. The resulting multiple prompts will be fed to LLMs. Then, multiple task-specific embeddings will be extracted.

Jiang et al. (2022a) incorporates in-context learning (Dong et al., 2023) to enhance sentence embeddings. While proven effective, it also reveals that the produced embeddings are task-specific and struggle with generalization across various downstream tasks, in addition to being highly sensitive to the choice of demonstrations.

3 Method

In this section, we begin by reviewing two kinds of previous prompting methods for deriving sentence representation from masked and causal language models, respectively (see Section 3.1). Subsequently, we describe our proposed method, i.e., **Meta**-Task Prompting with **E**xplicit **O**ne-Word **L**imitation (MetaEOL) (see Section 3.2) in detail. Lastly, we describe meta-tasks involved in this paper (see Section 3.3).

3.1 Previous Language Model Prompting

3.1.1 Masked Language Model

Masked language models, e.g., BERT (Devlin et al., 2019) and RoBERTa (Liu et al., 2019), use a mask prediction task to capture contextual information for a certain token. To align with this point, PromptBERT (Jiang et al., 2022a) formulates the sentence embedding extraction as a similar task and employs the following template,

This sentence : "[TEXT]" means [MASK] .

for prompting. Here, [TEXT] and *[MASK]* indicate the placeholder for the input sentence and the mask token. The last layer's hidden vector of *[MASK]* token is directly used as the sentence representation.

Jiang et al. (2022a) empirically show that such a simple prompting method can achieve decent performance, and equipping it with a contrastive loss for large-scale continued training leads to further enhancements for embedding quality. However, it is worth noting that extra training is resource-intensive, especially for today's LLMs. To enhance clarity, we provide results both with and without training on BERT and RoBERTa in the following experiments.

3.1.2 Causal Language Model

There are trials to directly extract sentence representation from large Causal Language Models (CLMs), e.g., OPT (Zhang et al., 2022) or LLAMA (Touvron et al., 2023a), without additional training. Inspired by (Jiang et al., 2022a), PromptEOL (Jiang et al., 2023b) employs a similar template as follows,

This sentence: " [TEXT] " means in one word: "

for prompting, where the last layer's hidden vector for the last token """ is extracted as the sentence representation. A constraint of "in one word" is applied to avoid the model's tendency to generate long sentences such that the last token fails to capture the overall information.

Nevertheless, the obtained embedding highly relies on the single prompt. So the inference process is confined and results in incomprehensive features, making it not applicable to extensive scenarios. As shown in Figure 1, for a negative review of a movie, the resulting embedding only contains the general semantics of movies but falls short of capturing other critical aspects such as sentiment.

3.2 Meta-Task Prompting

To overcome the issues raised above, we propose **Meta**-Task Prompting with **E**xplicit **O**ne-Word **L**imitation (MetaEOL). A meta-task is associated with a potential broad usage scenario for the corresponding sentence representation. As shown in Figure 2, we directly prompt casual LLMs with the goals of multiple meta tasks, aiming to obtain the representations under various broad intents.

Specifically, we produce task-oriented prompts by decorating the original prompting template used for causal LLMs (see Section 3.1.2) with the corresponding task description. For example, given a meta-task where representations are extracted for Text Classification (TC), we extend the template with task-oriented context to define the behavior during inference. As the template of *Meta Task-1* shown in Figure 2, a detailed task description text, telling the LLM that it should categorize the excerpt into a broad category, is placed at the beginning of the prompt. Then, an instruction with a constraint of "in one word" is followed to ensure models aggregate the information of the whole sentence into the embedding of the last token. The placeholder *[TEXT]* will be substituted with the original sentence to produce the final task-oriented prompt. The resulting task-specific prompt will serve as input to LLMs. Subsequently, we extract the hidden vector of the last token "“”" as the sentence representation, following the pattern outlined in Section 3.1.

It is worth noting that given various meta-tasks, distinct templates will be employed, leading to multiple different sentence embeddings. Our hypothesis posits that each embedding captures a distinct representation customized for a specific feature viewpoint (meta-task). In this paper, we empirically show that simply averaging different embedding derived from multiple meta-tasks can achieve superior performance for both intrinsic and downstream evaluation benchmarks.

3.3 Types of Meta-Tasks

In this paper, we conduct experiments on the following four distinct meta-tasks, i.e., Text Classification (TC), Sentiment Analysis (SA), Paraphrase Identification (PI), and Information Extraction (IE), aiming to capture information from different angles. E.g., intuitively, the TC task primarily emphasizes topic-level information, whereas the IE task concentrates on surface-level signals.

For each meta-task, we straightforwardly leverage ChatGPT-4 as a template generator to produce multiple templates. The instruction we used to prompt the ChatGPT-4 is provided in Appendix A.1.

Note that introducing more meta-tasks is trivial, which only requires adding more task names to the generator. Here, we choose the above four meta-tasks as a testbed to assess the scalability. More specifically, in Section 5.2, we show that incrementally adding more meta-tasks to our workflow results in consistently better performance.

4 Experiments

4.1 Settings

Dataset. Suggested by prior works (Reimers and Gurevych, 2019; Gao et al., 2021; Jiang et al., 2022b) that the primary objective of sentence embeddings is to cluster semantically similar sentences, we evaluate MetaEOL on seven semantic textual similarity (STS) datasets, utilizing the SentEval toolkit (Conneau and Kiela, 2018). The STS datasets consist of STS 2012-2016 (Agirre et al., 2012b, 2013, 2014, 2015, 2016), STS-B (Cer et al., 2017a), and SICK-R (Marelli et al., 2014). Each sentence pair in the STS datasets is annotated with a score from 0 to 5 indicating the pairwise semantic similarity. The Spearman correlation (scaled by 100x) between the model-predicted similarity scores and human-annotated similarity scores is used as the metric. We employ cosine similarity to measure the similarity between sentence embeddings. The Spearman correlation is computed under the "all" setting.

Baselines. The baselines we consider can be mainly categorized into two types – models with contrastive training and without contrastive training: (I) *Models with Contrastive Training:* We compare MetaEOL with SOTA unsupervised contrastive-trained models, namely SimCSE (Gao et al., 2021) and PromptBERT (Jiang et al., 2022a). The models are trained on 10^6 sentences randomly sampled from Wikipedia. Results based on BERT (Devlin et al., 2019) and RoBERTa (Liu et al., 2019) models are reported. And, (II) *Models without Contrastive Training:* We compare MetaEOL with (1) Average pooling methods, where average pooling is applied to the output hidden states of all tokens in a sentence to obtain the sentence embedding. We report results with BERT, the encoder of ST5 (Ni et al.,

Method	Params	STS12	STS13	STS14	STS15	STS16	STS-B	SICK-R	Avg.
Unsupervised Contrastive Training									
SimCSE-BERT	110M	68.40	82.41	74.38	80.91	78.56	76.85	72.23	76.25
SimCSE-RoBERTa	123M	70.16	81.77	73.24	81.36	80.65	80.22	68.56	76.57
PromptBERT	110M	71.56	84.58	76.98	84.47	80.60	81.60	69.87	78.54
PromptRoBERTa	123M	73.94	84.74	77.28	84.99	81.74	81.88	69.50	79.15
Without Contrastive Training									
BERT avg.	110M	30.87	59.89	47.73	60.29	63.73	47.29	58.22	52.57
BERT prompt	110M	60.96	73.83	62.18	71.54	68.68	70.60	67.16	67.85
ST5-Enc avg.	4.8B	34.97	60.19	47.59	66.40	70.62	62.83	63.57	58.02
LLAMA2 avg.	7B	35.49	53.15	40.12	55.35	53.26	42.10	49.96	47.06
Mistral avg.	7B	41.13	54.08	43.99	56.94	53.80	42.99	52.32	49.32
PromptEOL-LLAMA2	7B	58.81	77.01	66.34	73.22	73.56	71.66	69.64	70.03
PromptEOL-Mistral	7B	63.08	78.58	69.40	77.92	79.01	75.77	69.47	73.32
PromptEOL-LLAMA2	13B	56.19	76.42	65.42	7273	75.21	67.96	68.23	68.83
MetaEOL-LLAMA2 (***Ours***)	7B	64.16	81.61	73.09	81.11	78.94	77.96	74.86	75.96 (+5.93)
MetaEOL-Mistral (***Ours***)	7B	64.05	82.35	71.57	81.36	79.85	78.29	75.13	76.09 (+2.77)
MetaEOL-LLAMA2 (***Ours***)	13B	61.07	82.53	73.30	80.99	79.14	77.11	74.77	75.56 (+6.73)

Table 1: Results on STS tasks (Spearman correlation scaled by 100x). Values in parentheses, such as "(+5.93)" in MetaEOL's results, represent the increase in average score compared to the average score of the same model utilizing PromptEOL.

2022), LLAMA2 (Touvron et al., 2023b) and Mistral (Jiang et al., 2023a) models; and (2) Prompt-based methods, which include BERT Prompt that employs the same prompt strategy as PromptBERT but does not incorporate contrastive training, and also PromptEOL. All methods mentioned above rely on the output from the final layer to obtain the sentence embedding.

Implementation Details. We apply MetaEOL to LLaMA2-7B, LLAMA2-13B, and Mistral-7B models, using meta-tasks consisting of Text Classification (TC), Sentiment Analysis (SA), Paraphrase Identification (PI), and Information Extraction (IE). These tasks are distinct and collectively consider diverse aspects of a sentence. For each of these meta-tasks, we utilize GPT-4 to create two unique task prompts, resulting in a total of eight task prompts.[2] MetaEOL rely on the output from the final layer to obtain the sentence embedding. We simply average the resulting embeddings of task prompts from different meta-tasks to obtain the final embedding.

4.2 Main Results

The results of MetaEOL on STS tasks are shown in Table 1, with a notable performance by MetaEOL which requires no training. Among models that do not require training, prompt-based methods exhibit superior results compared to average pooling methods, especially with the LLAMA and Mistral models. Across various models including LLAMA2-7B/13B and Mistral-7B, MetaEOL, which does not require any training, demonstrates competitive performance compared to contrastive-trained models such as SimCSE-BERT and SimCSE-Roberta, albeit with a slight lag behind PromptBERT. Furthermore, MetaEOL significantly outperforms PromptEOL across three test models, demonstrating a consistent improvement. Notably, the LLAMA2-13B model using MetaEOL shows an average improvement of 6.73% over PromptEOL, underscoring the efficacy of MetaEOL.

4.3 Qualitative Example

We further show the top-10 tokens predicted by different task prompts in Table 2. The example illustrates that PromptEOL creates sentence embeddings focusing on stop-word tokens (such as *a*, *this*, *the*, *it*), which convey minimal information. In contrast, the four meta-tasks of MetaEOL demonstrably shift the behavior of the embeddings, leading to the prediction of tokens that are distinct and imbued with substantive meaning.

Specifically, Text Classification steers the embeddings toward tokens that are indicative of specific topics, such as *Culture*. Sentiment Analysis is inclined to produce embeddings close to sentiment-related words. Paraphrase Identification yields em-

[2]The details of these eight task prompts are presented in Appendix A.3.

Sentence	Prompt	Top-predicted tokens
Smart and alert, thirteen conversations about one thing is a small gem.	PromptEOL	I one a thing the This The smart It it
	Text Classification	Culture E Pol \n Bus " Culture educ Te Health
	Sentiment Analysis	positive pos good ext good very neut negative smart extremely
	Paraphrase Identification	smart a the intelligent The short clever conc A conversation
	Information Extraction	gem smart thing alert small conversation Gem thirteen gem a

Table 2: The top-10 tokens predicted by different task prompts with Mistral-7B. PromptEOL creates sentence embeddings with an emphasis on stop-word tokens. Text Classification focuses embeddings on topic-relevant tokens like *Culture*. Sentiment Analysis aligns embeddings with sentiment words. Paraphrase Identification diversifies embeddings with synonyms, adding richness with terms like *intelligent*, *short*, and *clever*. Information Extraction steers embeddings toward key factual tokens.

Method	STS Avg.
PromptEOL	70.03
w. 8 paraphrases	62.72
MetaEOL	75.96
TC only	70.92
SA only	67.06
PI only	73.03
IE only	72.06
w. embedding concatenation	74.99

Table 3: Ablation study on LLAMA2-7B. STS Avg. refers to the average score of the seven STS tasks. TC: Text Classification; SA: Sentiment Analysis; PI: Paraphrase Identification; IE: Information Extraction.

beddings that capture a spectrum of synonyms, enriching the sentence with varied linguistic expressions like *intelligent*, *short*, and *clever*. Information Extraction modifies the embeddings towards tokens that represent key facts or elements within the sentence.

5 Analysis

In this section, we thoroughly analyze MetaEOL using the LLAMA2-7B model.

5.1 Ablation Study

We evaluate the effectiveness of key components of MetaEOL in table 3. First, to ensure the improvement observed with MetaEOL is not merely due to involving more prompts, we create seven paraphrased versions of the PromptEOL prompt, resulting in a total of eight prompts.[3] We then average the embeddings from these eight prompts to form the final sentence embedding. We find merely duplicating PromptEOL prompts (*w.* 8 paraphrase) does not improve PromptEOL but results in a significant decline. Additionally, we implement MetaEOL exclusively on each meta-task (TC/SA/PI/IE only). We find that tasks requiring a detailed comprehension of sentences (PI and IE) yield superior performance compared to those requiring a broader understanding, even surpassing PromptEOL. However, MetaEOL, which combines the embeddings from these meta-tasks, outperforms all individual meta-tasks, confirming the complementarity of the meta-tasks and the effectiveness of combining embeddings from diverse meta-tasks. We finally find that averaging the embeddings from different meta-tasks yields better results than concatenating them.

5.2 Influence of Number of Tasks

We investigate the influence of the number of tasks as presented in table 4. We find increasing the number of tasks leads to a consistent improvement in performance on average and nearly every individual STS task. This further verifies the complementarity of the meta-tasks and underscores the importance of utilizing various diverse meta-tasks.

5.3 Influence of Number of Prompts

Here, we investigate the impact of the number of prompts in Figure 3. We concentrate on Sentiment Analysis as the meta-task and utilize GPT-4 to generate three additional Sentiment Analysis prompts besides the two we used in MetaEOL. This results in a total of five distinct prompts, specifically tailored for Product Review Rating, Emotion Detection, Sentiment Polarity Detection, Sentiment Intensity and Emotion Detection, and Aspect-Based Sentiment Analysis, respectively.[4] We systematically computed the average performance across all combinations of these five prompts, conditioned on a fixed number of prompts.

As Figure 3 shows, increasing the number of prompts within a particular task type facilitates more nuanced embeddings, thereby leading to better STS results. We opt for two prompts for each

[3]The seven paraphrased prompts are presented in Appendix A.2

[4]The details of these five instructions are in Appendix A.4.

Meta-Tasks	STS12	STS13	STS14	STS15	STS16	STS-B	SICK-R	Avg.
TC	58.36	75.57	67.20	77.04	74.51	71.84	71.90	70.92
TC+SA	58.89	75.56	67.35	77.60	74.90	73.58	72.48	71.48
TC+SA+PI	63.08	80.01	71.24	80.38	78.26	77.42	75.00	75.06
TC+SA+PI+IE	64.16	81.61	73.09	81.11	78.94	77.96	74.86	75.96

Table 4: Results on increasing number of tasks with LLAMA2-7B. TC: Text Classification; SA: Sentiment Analysis; PI: Paraphrase Identification; IE: Information Extraction.

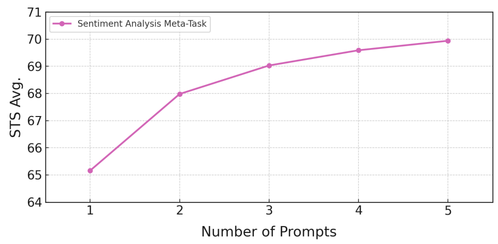

Figure 3: Influence of number of prompts on LLAMA2-7B. STS Avg. refers to the average score of the seven STS tasks.

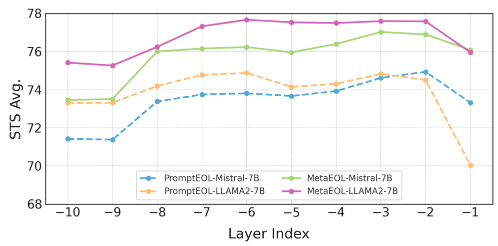

Figure 4: Influence of output layer index. STS Avg. refers to the average score of the seven STS tasks.

Model	Layer Index	STS Avg.
LLAMA2-7B	-1	75.35
LLAMA2-13B	-1	74.96
LLAMA2-70B	-1	75.41
LLAMA2-7B	-3	77.00
LLAMA2-13B	-4	76.08
LLAMA2-70B	-8	78.06

Table 5: Results of MetaEOL on increasing the model size. All models are loaded with 4-bit precision. We develop a proportional layer selection strategy, leveraging the last 10% of layers to derive sentence embeddings (specifically, the third-to-last, fourth-to-last, and eighth-to-last layers for the 7B, 13B, and 70B models, respectively), and obtain the best results with the 70B model.

meta-task for MetaEOL to optimize both performance and computational efficiency.

5.4 Influence of Output Layers

We check the impact of output layers for LLAMA2 and Mistral-7B models, using PromptEOL and MetaEOL. Figure 4 presents the STS average scores across different output layer indices.

It is highlighted that the third-to-last layers (indexed as -3) across all four configurations perform similarly well, which suggests that this layer can be considered as a point of convergence in terms of optimal performance for these models.

MetaEOL outperforms PromptEOL across all layers and configurations. Interestingly, PromptEOL tends to show more variability in performance across different layers compared to MetaEOL. This suggests that the MetaEOL approach potentially stabilizes the representational quality across layers.

5.5 Scaling LLMs

In this section, we investigate the impact of model size on the performance of MetaEOL. For the sake of computational resources, we load models with 4-bit precision.

Informed by the insights observed from Section 5.4, which suggested that for 7B models, the layer index -3 can be considered optimal, as evidenced by its performance in both PromptEOL and MetaEOL. We, therefore, propose a simple proportional layer selection strategy, opting for layers -3 of 32, -4 of 40, and -8 of 80 as the output layers for the LLAMA2-7B, LLAMA2-13B, and LLAMA2-70B models respectively. This approach aligns with the model sizes, which correlates to 10% from the final layer.

The results show that using the final layer for sentence embedding generation, which is indicated by layer index -1, does not yield improved performance with increased model size. Contrastingly, the application of our proportional layer strategy reveals a different trend. Specifically, the LLAMA2-70B model, which utilizes the -8 layer, demonstrates superior performance, suggesting that larger models might benefit more significantly from selecting a proportionate layer rather than the last layer for sentence embedding. This observation could point to a potential scaling law, where larger models require a different, non-final layer to maximize performance effectively.

Method	Params	MR	CR	SUBJ	MPQA	SST	TREC	MRPC	Avg.
Fine-tuning on supervised datasets									
SimCSE-RoBERTa	123M	84.92	92.00	94.11	89.82	91.27	88.80	75.65	88.08
ST5-Enc	4.8B	90.83	94.44	96.33	91.68	94.84	95.40	77.91	91.63
Without fine-tuning									
MRPrompt-LLAMA2	7B	***91.82***	92.88	97.07	91.60	96.54	95.80	74.61	91.47
CRPrompt-LLAMA2	7B	91.17	***93.27***	96.62	91.75	96.60	95.80	73.22	91.20
SUBJPrompt-LLAMA2	7B	91.88	93.17	***96.96***	91.09	95.66	96.00	76.41	91.60
MPQAPrompt-LLAMA2	7B	91.10	93.04	96.30	***91.82***	95.72	96.00	75.42	91.34
SSTPrompt-LLAMA2	7B	91.82	92.88	97.07	91.60	***96.54***	95.80	74.61	91.47
TRECPrompt-LLAMA2	7B	88.97	92.19	96.23	91.45	94.18	***96.80***	74.72	90.65
MRPCPrompt-LLAMA2	7B	90.33	93.32	96.36	91.45	94.67	96.00	***75.13***	91.04
								Avg. on task-specific prompting (i.e., diagonal):	***91.76***
PromptEOL-LLAMA2	7B	90.63	92.87	96.32	91.19	95.00	95.40	75.19	90.94
MetaEOL-LLAMA2 (***Ours***)	7B	90.93	93.51	96.12	91.95	95.77	97.60	76.81	91.81

Table 6: Results on transfer learning tasks. We design task-specific prompts for each task, denoted as {TASK}Prompt where {TASK} is a placeholder for the task's name. The corresponding task performance of each specific prompt and their average is ***bold italic***. SST and MR share the same prompt. These task-specific prompts can significantly improve the performance of the corresponding tasks compared to both PromptEOL and ST5-Enc. MetaEOL yields superior results even without being explicitly customized for these tasks.

5.6 Transfer Learning Tasks

We conclude our analysis by assessing the performance of MetaEOL on transfer learning tasks. Following prior works (Gao et al., 2021; Ni et al., 2022), we utilize the standard transfer learning tasks provided by SentEval. The tasks consist of MR (Pang and Lee, 2005), CR (Hu and Liu, 2004), SUBJ (Pang and Lee, 2004), MPQA (Wiebe et al., 2005), SST-2 (Socher et al., 2013), TREC (Voorhees and Tice, 2000), and MRPC (Dolan and Brockett, 2005). We include two supervised contrastive-trained models (SimCSE and ST5-Enc) for reference. Notably, ST5-Enc, a model with a 4.8B parameter count, is extensively trained on natural language inference (NLI) data and two billion question-answer pairs.

To investigate the ability of task-specific prompts to modify embedding behavior, we have crafted task prompts tailored to each SentEval task.[5] As an example, for the Movie Review (MR) dataset, we designed a prompt structured as: *In this task, you're given a movie review, and you need to classify its sentiment into positive or negative. For this task, this sentence: "input sentence" means in one word:"*, referred to as MRPrompt in Table 6. These task-specific prompts significantly improve the corresponding task performance, always better than PromptEOL and heavily supervised contrastive-trained ST5-Enc, verifying that LLAMA2-7B can follow the prompt to generate tailored embeddings without any training. This indicates that carefully designed prompts can effectively steer the pre-trained embeddings to align with various NLP tasks, thus providing a more resource-efficient alternative to the traditional fine-tuning paradigm.

Moreover, although without being explicitly customized for these tasks, MetaEOL achieves the highest average result, even outperforming heavily trained ST5-Enc. This suggests that the integration of the four meta-tasks in MetaEOL can cultivate generalized embeddings that perform admirably across different tasks.

6 Conclusion

In this paper, we introduce MetaEOL, a new approach for deriving high-quality sentence embeddings from LLMs without requiring any training. By leveraging a diverse set of meta-task prompts, MetaEOL effectively captures multiple representations of sentences from distinct perspectives. We show simply averaging these meta-task derived embeddings leads to generalized general-purpose embeddings, which work remarkably well across STS datasets and transfer learning tasks.

[5]The details of the task prompts are presented in Appendix A.5.

Limitations

We acknowledge two limitations in our work: computational overhead and restricted evaluation benchmarks. As MetaEOL requires feeding multiple prompts to LLMs to generate several embeddings, the computational cost will be higher than previous methods. Nonetheless, in contexts where sentences are consistently reused, such as when embeddings are stored for downstream classification or retrieval tasks, this issue becomes less significant. Furthermore, our evaluation is currently confined to sentence-level tasks in English only. As LLMs continue to advance, exploring the performance of MetaEOL in multilingual contexts and its applicability to document retrieval presents an intriguing avenue for future research.

References

Eneko Agirre, Carmen Banea, Claire Cardie, Daniel Cer, Mona Diab, Aitor Gonzalez-Agirre, Weiwei Guo, Inigo Lopez-Gazpio, Montse Maritxalar, Rada Mihalcea, et al. 2015. Semeval-2015 task 2: Semantic textual similarity, english, spanish and pilot on interpretability. In *Proceedings of the 9th international workshop on semantic evaluation (SemEval 2015)*, pages 252–263.

Eneko Agirre, Carmen Banea, Claire Cardie, Daniel Cer, Mona Diab, Aitor Gonzalez-Agirre, Weiwei Guo, Rada Mihalcea, German Rigau, and Janyce Wiebe. 2014. Semeval-2014 task 10: Multilingual semantic textual similarity. In *Proceedings of the 8th international workshop on semantic evaluation (SemEval 2014)*, pages 81–91.

Eneko Agirre, Carmen Banea, Daniel Cer, Mona Diab, Aitor Gonzalez Agirre, Rada Mihalcea, German Rigau Claramunt, and Janyce Wiebe. 2016. Semeval-2016 task 1: Semantic textual similarity, monolingual and cross-lingual evaluation. In *SemEval-2016. 10th International Workshop on Semantic Evaluation; 2016 Jun 16-17; San Diego, CA. Stroudsburg (PA): ACL; 2016. p. 497-511.* ACL (Association for Computational Linguistics).

Eneko Agirre, Daniel Cer, Mona Diab, and Aitor Gonzalez-Agirre. 2012a. SemEval-2012 task 6: A pilot on semantic textual similarity. In **SEM 2012: The First Joint Conference on Lexical and Computational Semantics – Volume 1: Proceedings of the main conference and the shared task, and Volume 2: Proceedings of the Sixth International Workshop on Semantic Evaluation (SemEval 2012)*, pages 385–393, Montréal, Canada. Association for Computational Linguistics.

Eneko Agirre, Daniel Cer, Mona Diab, and Aitor Gonzalez-Agirre. 2012b. Semeval-2012 task 6: A pilot on semantic textual similarity. In ** SEM 2012: The First Joint Conference on Lexical and Computational Semantics–Volume 1: Proceedings of the main conference and the shared task, and Volume 2: Proceedings of the Sixth International Workshop on Semantic Evaluation (SemEval 2012)*, pages 385–393.

Eneko Agirre, Daniel Cer, Mona Diab, Aitor Gonzalez-Agirre, and Weiwei Guo. 2013. Sem 2013 shared task: Semantic textual similarity. In *Second joint conference on lexical and computational semantics (* SEM), volume 1: proceedings of the Main conference and the shared task: semantic textual similarity*, pages 32–43.

Akari Asai, Timo Schick, Patrick Lewis, Xilun Chen, Gautier Izacard, Sebastian Riedel, Hannaneh Hajishirzi, and Wen-tau Yih. 2023. Task-aware retrieval with instructions. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 3650–3675, Toronto, Canada. Association for Computational Linguistics.

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.

Daniel Cer, Mona Diab, Eneko Agirre, Inigo Lopez-Gazpio, and Lucia Specia. 2017a. Semeval-2017 task 1: Semantic textual similarity-multilingual and cross-lingual focused evaluation. *arXiv preprint arXiv:1708.00055*.

Daniel Cer, Mona Diab, Eneko Agirre, Iñigo Lopez-Gazpio, and Lucia Specia. 2017b. SemEval-2017 task 1: Semantic textual similarity multilingual and crosslingual focused evaluation. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 1–14, Vancouver, Canada. Association for Computational Linguistics.

Qinyuan Cheng, Xiaogui Yang, Tianxiang Sun, Linyang Li, and Xipeng Qiu. 2023. Improving contrastive learning of sentence embeddings from AI feedback. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 11122–11138, Toronto, Canada. Association for Computational Linguistics.

Yung-Sung Chuang, Rumen Dangovski, Hongyin Luo, Yang Zhang, Shiyu Chang, Marin Soljacic, Shang-Wen Li, Scott Yih, Yoon Kim, and James Glass. 2022. DiffCSE: Difference-based contrastive learning for sentence embeddings. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4207–4218, Seattle, United States. Association for Computational Linguistics.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi

Wang, Mostafa Dehghani, Siddhartha Brahma, Albert Webson, Shixiang Shane Gu, Zhuyun Dai, Mirac Suzgun, Xinyun Chen, Aakanksha Chowdhery, Alex Castro-Ros, Marie Pellat, Kevin Robinson, Dasha Valter, Sharan Narang, Gaurav Mishra, Adams Yu, Vincent Zhao, Yanping Huang, Andrew Dai, Hongkun Yu, Slav Petrov, Ed H. Chi, Jeff Dean, Jacob Devlin, Adam Roberts, Denny Zhou, Quoc V. Le, and Jason Wei. 2022. Scaling instruction-finetuned language models. *arXiv preprint arXiv:2210.11416*.

Alexis Conneau and Douwe Kiela. 2018. SentEval: An evaluation toolkit for universal sentence representations. In *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC 2018)*, Miyazaki, Japan. European Language Resources Association (ELRA).

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

William B Dolan and Chris Brockett. 2005. Automatically constructing a corpus of sentential paraphrases. In *Proceedings of the Third International Workshop on Paraphrasing (IWP2005)*.

Qingxiu Dong, Lei Li, Damai Dai, Ce Zheng, Zhiyong Wu, Baobao Chang, Xu Sun, Jingjing Xu, Lei Li, and Zhifang Sui. 2023. A survey on in-context learning. *arXiv preprint arXiv:2301.00234*.

Tianyu Gao, Xingcheng Yao, and Danqi Chen. 2021. Simcse: Simple contrastive learning of sentence embeddings. *arXiv preprint arXiv:2104.08821*.

Yun He, Steven Zheng, Yi Tay, Jai Gupta, Yu Du, Vamsi Aribandi, Zhe Zhao, Yaguang Li, Zhao Chen, Donald Metzler, Heng-Tze Cheng, and Ed H. Chi. 2022. HyperPrompt: Prompt-based task-conditioning of transformers. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pages 8678–8690. PMLR.

Minqing Hu and Bing Liu. 2004. Mining and summarizing customer reviews. In *ACM SIGKDD international conference on Knowledge discovery and data mining*.

Gautier Izacard, Mathilde Caron, Lucas Hosseini, Sebastian Riedel, Piotr Bojanowski, Armand Joulin, and Edouard Grave. 2021. Unsupervised dense information retrieval with contrastive learning. *arXiv preprint arXiv:2112.09118*.

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. 2023a. Mistral 7b. *arXiv preprint arXiv:2310.06825*.

Ting Jiang, Shaohan Huang, Zhongzhi Luan, Deqing Wang, and Fuzhen Zhuang. 2023b. Scaling sentence embeddings with large language models. *arXiv preprint arXiv:2307.16645*.

Ting Jiang, Jian Jiao, Shaohan Huang, Zihan Zhang, Deqing Wang, Fuzhen Zhuang, Furu Wei, Haizhen Huang, Denvy Deng, and Qi Zhang. 2022a. Promptbert: Improving bert sentence embeddings with prompts. *arXiv preprint arXiv:2201.04337*.

Ting Jiang, Jian Jiao, Shaohan Huang, Zihan Zhang, Deqing Wang, Fuzhen Zhuang, Furu Wei, Haizhen Huang, Denvy Deng, and Qi Zhang. 2022b. PromptBERT: Improving BERT sentence embeddings with prompts. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 8826–8837, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Pei Ke, Haozhe Ji, Siyang Liu, Xiaoyan Zhu, and Minlie Huang. 2020. SentiLARE: Sentiment-aware language representation learning with linguistic knowledge. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6975–6988, Online. Association for Computational Linguistics.

Tian Yu Liu, Matthew Trager, Alessandro Achille, Pramuditha Perera, Luca Zancato, and Stefano Soatto. 2023a. Meaning representations from trajectories in autoregressive models. *arXiv preprint arXiv:2310.18348*.

Tian Yu Liu, Matthew Trager, Alessandro Achille, Pramuditha Perera, Luca Zancato, and Stefano Soatto. 2023b. Meaning representations from trajectories in autoregressive models. *arXiv preprint arXiv:2310.18348*.

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*.

Marco Marelli, Stefano Menini, Marco Baroni, Luisa Bentivogli, Raffaella Bernardi, Roberto Zamparelli, et al. 2014. A sick cure for the evaluation of compositional distributional semantic models. In *Lrec*, pages 216–223. Reykjavik.

Swaroop Mishra, Daniel Khashabi, Chitta Baral, and Hannaneh Hajishirzi. 2022. Cross-task generalization via natural language crowdsourcing instructions. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3470–3487, Dublin, Ireland. Association for Computational Linguistics.

Bhaskar Mitra, Fernando Diaz, and Nick Craswell. 2017. Learning to match using local and distributed representations of text for web search. In *Proceedings of the 26th international conference on world wide web*, pages 1291–1299.

Jianmo Ni, Gustavo Hernandez Abrego, Noah Constant, Ji Ma, Keith Hall, Daniel Cer, and Yinfei Yang. 2022. Sentence-t5: Scalable sentence encoders from pre-trained text-to-text models. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 1864–1874, Dublin, Ireland. Association for Computational Linguistics.

Bo Pang and Lillian Lee. 2004. A sentimental education: Sentiment analysis using subjectivity summarization based on minimum cuts. In *Proceedings of the 42nd Annual Meeting of the Association for Computational Linguistics (ACL-04)*, pages 271–278, Barcelona, Spain.

Bo Pang and Lillian Lee. 2005. Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales. In *Proceedings of the ACL*.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3982–3992, Hong Kong, China. Association for Computational Linguistics.

Victor Sanh, Albert Webson, Colin Raffel, Stephen H. Bach, Lintang Sutawika, Zaid Alyafeai, Antoine Chaffin, Arnaud Stiegler, Teven Le Scao, Arun Raja, Manan Dey, M Saiful Bari, Canwen Xu, Urmish Thakker, Shanya Sharma Sharma, Eliza Szczechla, Taewoon Kim, Gunjan Chhablani, Nihal Nayak, Debajyoti Datta, Jonathan Chang, Mike Tian-Jian Jiang, Han Wang, Matteo Manica, Sheng Shen, Zheng Xin Yong, Harshit Pandey, Rachel Bawden, Thomas Wang, Trishala Neeraj, Jos Rozen, Abheesht Sharma, Andrea Santilli, Thibault Fevry, Jason Alan Fries, Ryan Teehan, Tali Bers, Stella Biderman, Leo Gao, Thomas Wolf, and Alexander M. Rush. 2022. Multitask prompted training enables zero-shot task generalization. *arXiv preprint arXiv:2110.08207*.

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D. Manning, Andrew Ng, and Christopher Potts. 2013. Recursive deep models for semantic compositionality over a sentiment treebank. In *emnlp*, pages 1631–1642.

Hongjin Su, Weijia Shi, Jungo Kasai, Yizhong Wang, Yushi Hu, Mari Ostendorf, Wen-tau Yih, Noah A. Smith, Luke Zettlemoyer, and Tao Yu. 2023. One embedder, any task: Instruction-finetuned text embeddings. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 1102–1121, Toronto, Canada. Association for Computational Linguistics.

Michael Tomasello. 2009. The usage-based theory of language acquisition. In *The Cambridge handbook of child language*, pages 69–87. Cambridge Univ. Press.

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, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. 2023b. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Ellen M Voorhees and Dawn M Tice. 2000. Building a question answering test collection. In *Proceedings of the 23rd annual international ACM SIGIR conference on Research and development in information retrieval*, pages 200–207.

Yizhong Wang, Swaroop Mishra, Pegah Alipoormolabashi, Yeganeh Kordi, Amirreza Mirzaei, Atharva Naik, Arjun Ashok, Arut Selvan Dhanasekaran, Anjana Arunkumar, David Stap, Eshaan Pathak, Giannis Karamanolakis, Haizhi Lai, Ishan Purohit, Ishani Mondal, Jacob Anderson, Kirby Kuznia, Krima Doshi, Kuntal Kumar Pal, Maitreya Patel, Mehrad Moradshahi, Mihir Parmar, Mirali Purohit, Neeraj Varshney, Phani Rohitha Kaza, Pulkit Verma, Ravsehaj Singh Puri, Rushang Karia, Savan Doshi, Shailaja Keyur Sampat, Siddhartha Mishra, Sujan Reddy A, Sumanta Patro, Tanay Dixit, and Xudong Shen. 2022. Super-NaturalInstructions: Generalization via declarative instructions on 1600+ NLP tasks. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 5085–5109, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Janyce Wiebe, Theresa Wilson, and Claire Cardie. 2005. Annotating expressions of opinions and emotions in language. *Language resources and evaluation*, 39(2-3):165–210.

Qiyu Wu, Chongyang Tao, Tao Shen, Can Xu, Xiubo Geng, and Daxin Jiang. 2022. PCL: Peer-contrastive learning with diverse augmentations for unsupervised sentence embeddings. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 12052–12066, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Junlei Zhang, Zhenzhong Lan, and Junxian He. 2023. Contrastive learning of sentence embeddings from scratch. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 3916–3932, Singapore. Association for Computational Linguistics.

Susan Zhang, Stephen Roller, Naman Goyal, Mikel Artetxe, Moya Chen, Shuohui Chen, Christopher Dewan, Mona Diab, Xian Li, Xi Victoria Lin, Todor Mihaylov, Myle Ott, Sam Shleifer, Kurt Shuster, Daniel Simig, Punit Singh Koura, Anjali Sridhar, Tianlu Wang, and Luke Zettlemoyer. 2022. Opt: Open pre-trained transformer language models. *arXiv preprint arXiv:2205.01068*.

A Appendix

A.1 Instruction to Prompt ChatGPT4 for Template Generation

We insert a blank line between paragraphs to enhance readability.

> Obtaining the representation of sentences is a fundamental task in natural language processing.
>
> The representation can not only be used to compute the semantic similarity between different sentences but also to be directly used for downstream tasks, like Text Categorization, Sentiment Analysis, Summarization, Style Transfer, Text Simplification, and Sentence Composition.
>
> A common way to obtain the representation is to use the format "This sentence "input sentence" means in one word:"" and use the hidden states of the last token as the representation of the sentence. However, we want a versatile representation that covers various aspects of the sentence by adding task instructions before the format. For instance: "In this task, you're given a review from Amazon. Your task is to generate a rating for the product on a scale of 1-5 based on the review. The rating means 1: extremely poor, 2: poor, 3: neutral, 4: good, 5: extremely good. For this task, this sentence : "input sentence" means in one word:"" is used to obtain the representation of the sentence conditioned on the given task.
>
> Can you help me write task instructions that can cover different aspects of the sentence such that the representation is versatile to both similarity tasks and downstream tasks?
>
> Please write two instructions for each of the Text Classification, Sentiment Analysis, Paraphrase Identification, and Information Extraction tasks.

A.2 Paraphrased Prompts of PromptEOL

> This sentence : "input sentence" can be rephrased to one word:"
> This sentence : "input sentence" can be expressed as one word:"
> This sentence : "input sentence" implies in one word:"
> This sentence : "input sentence" indicates in one word:"
> The meaning of this sentence : "input sentence" can be conveyed in another word:"
> This sentence : "input sentence" can be restated as one word:"
> This sentence : "input sentence" can be reformulated as one word:"

A.3 Prompts of MetaEOL

Text Classification

General Category Identification: In this task, you're presented with a text excerpt. Your task is to categorize the excerpt into a broad category such as 'Education', 'Technology', 'Health', 'Business', 'Environment', 'Politics', or 'Culture'. These categories help in organizing content for better accessibility and targeting. For this task, this sentence : "input sentence" should be classified under one general category in one word:"

Opinion vs. Fact Discrimination: In this task, you're given a statement and you need to determine whether it's presenting an 'Opinion' or a 'Fact'. This distinction is vital for information verification, educational purposes, and content analysis. For this task, this sentence : "input sentence" discriminates between opinion and fact in one word:"

Sentiment Analysis

Product Review Rating: In this task, you're given a review from an online platform. Your task is to generate a rating for the product based on the review on a scale of 1-5, where 1 means 'extremely negative' and 5 means 'extremely positive'. For this task, this sentence : "input sentence" reflects the sentiment in one word:"

Emotion Detection: In this task, you're reading a personal diary entry. Your task is to identify the predominant emotion expressed, such as joy, sadness, anger, fear, or love. For this task, this sentence : "input sentence" conveys the emotion in one word:"

Paraphrase Identification

Similarity Check: In this task, you're presented with two sentences. Your task is to assess whether the sentences convey the same meaning. Use 'identical', 'similar', 'different', or 'unrelated' to describe the relationship. To enhance the performance of this task, this sentence : "input sentence" means in one word:"

Contextual Synonym Detection: In this task, you're given a sentence and a phrase. Your task is to determine if the phrase can be a contextual synonym within the given sentence. Options include 'yes', 'no', or 'partially'. To enhance the performance of this task, this sentence : "input sentence" means in one word:"

Information Extraction

Key Fact Identification: In this task, you're examining a news article. Your task is to extract the most critical fact from the article. For this task, this sentence : "input sentence" encapsulates the key fact in one word:"

Entity and Relation Extraction: In this task, you're reviewing a scientific abstract. Your task is to identify the main entities (e.g., proteins, diseases) and their relations (e.g., causes, treats). For this task, this sentence : "input sentence" highlights the primary entity or relation in one word:"

A.4 Prompts of the Sentiment Analysis Meta-Task

Sentiment Analysis Meta-Task
Product Review Rating: In this task, you're given a review from an online platform. Your task is to generate a rating for the product based on the review on a scale of 1-5, where 1 means 'extremely negative' and 5 means 'extremely positive'. For this task, this sentence : "input sentence" reflects the sentiment in one word:"
Emotion Detection: In this task, you're reading a personal diary entry. Your task is to identify the predominant emotion expressed, such as joy, sadness, anger, fear, or love. For this task, this sentence : "input sentence" conveys the emotion in one word:"
Sentiment Polarity Detection: In this task, you're analyzing customer feedback from various platforms. Your task is to identify the overall sentiment polarity of the feedback. The sentiment polarity means: 1 for very negative, 2 for negative, 3 for neutral, 4 for positive, and 5 for very positive. Based on this guidance, this sentence : "input sentence" represents in one word:"
Sentiment Intensity and Emotion Detection: In this task, your objective is to gauge the intensity and type of emotion conveyed in a piece of text, such as a social media post or a product review. This involves not just identifying whether the sentiment is positive or negative, but also understanding the strength of that sentiment and the specific emotions involved (e.g., joy, anger, sadness, surprise). For this task, this sentence : "input sentence" conveys an emotion that is best described in one word as:"
Aspect-based Sentiment Analysis: In this task, you're given a review of a product or service. Your task is to assess the sentiment toward specific aspects of the product or service mentioned in the review. For each mentioned aspect (e.g., quality, price, customer service), classify the sentiment as: 1 for very negative, 2 for negative, 3 for neutral, 4 for positive, and 5 for very positive. Based on this instruction, this sentence : "input sentence" signifies in one word:"

A.5 Task-Specific Prompts on Transfer Tasks

MR/SST
In this task, you're given a movie review, and you need to classify its sentiment into positive or negative. For this task, this sentence : "input sentence" means in one word:"

CR
In this task, you're given a customer review of a product sold online, and you need to classify its sentiment into positive or negative. For this task, this sentence : "input sentence" means in one word:"

SUBJ
In this task, you're analyzing movie reviews to determine their level of subjectivity. A subjective review is filled with personal opinions, feelings, and preferences of the reviewer, often expressing likes or dislikes and personal experiences. An objective review, on the other hand, sticks to factual information, such as plot details or actor performances, without revealing the reviewer's personal stance. For this task, this sentence : "input sentence" means in one word:"

MPQA
In this task, you are given a description of a entity or event expressed in data such as blogs, newswire, and editorials. You need to classify its sentiment into positive or negative. For this task, this sentence : "input sentence" means in one word:"

TREC
In this task, you are given a question. You need to detect which category better describes the question. A question belongs to the description category if it asks about description and abstract concepts. Entity questions are about entities such as animals, colors, sports, etc. Abbreviation questions ask about abbreviations and expressions abbreviated. Questions regarding human beings, description of a person, and a group or organization of persons are categorized as Human. Quantity questions are asking about numeric values and Location questions ask about locations, cities, and countries. Answer with "Description", "Entity", "Abbreviation", "Person", "Quantity", and "Location". For this task, this sentence : "input sentence" means in one word:"

MRPC
In this task, you are given two sentences(Sentence1 and Sentence2). Answer "Yes" if these sentences are a paraphrase of one another, otherwise answer "No". For this task, this sentence : "input sentence" means in one word:"