of next word $P(x_t|[\boldsymbol{x}_{<t}; \boldsymbol{x}_{\text{diagnosis}}]; \theta)$ by appending self-diagnosis textual input to the original input as mentioned above. These two probability distributions for the next token can be combined to suppress the undesired attribute.

Dataset Construction Schick and Schütze (2021) propose to use pre-trained LMs to generate datasets given certain instructions. As an example, suppose we have an unlabeled dataset in which each sample is a sentence. If we want to construct a dataset containing pairs of semantically similar sentences, then we can use the following template for each input sentence: "Write two sentences that mean the same thing. `[X] [Z]`" and attempt to generate a sentence that shares the same meaning as the input sentence.

8.10 Resources

We also collect some useful resources for different prompt-based applications.

Dataset Some datasets specifically designed for few-shot and zero-shot learning are shown in Tab. 9.

Task	Dataset	Setting	URL
Commonsense Reasoning	Pronoun Disambiguation Problems [93]	Zero	https://cs.nyu.edu/ davise/papers/...
	Winograd Schema Challenge [93]	Zero	https://cs.nyu.edu/ davise/papers/...
	CPRAG-102 [39]	Zero	https://github.com/aetting/lm-diagnostics
Linguistic Capacity Probing	WNLaMPro [150]	Zero	https://github.com/timoschick/...
	ROLE-88 [39]	Zero	https://github.com/aetting/lm-diagnostics
	NEG-136 [39]	Zero	https://github.com/aetting/lm-diagnostics
Fact Probing	LAMA [133]	Zero	https://dl.fbaipublicfiles.com/LAMA/...
	Negated LAMA [74]	Zero	https://github.com/norakassner/LAMA...
	Misprimed LAMA [74]	Zero	https://github.com/norakassner/LAMA...
	X-FACTR [66]	Zero	https://x-factr.github.io/
	LAMA-TREx-easy-hard [203]	Zero	https://github.com/princeton-nlp/...
Text Classification	FLEX [15]	Zero,Few	https://github.com/allenai/flex
	FewGLUE [154]	Few	https://github.com/timoschick/fewglue
General Conditional Gen.	REALTOXICITYPROMPTS [47]	Zero	https://allenai.org/data/...
	Natural-Instructions [120]	Few,Full	https://instructions.apps.allenai.org/

Table 9: Few-shot and zero-shot datasets for prompt-based learning.

Prompts As shown in Tab. 10, we collect existing commonly-used prompts designed manually, which can be regarded as off-the-shelf resource for future research and applications.

9 Prompt-relevant Topics

What is the essence of prompt-based learning and how does it relate to other learning methods? In this section, we connect prompt learning with other similar learning methods.

Ensemble Learning *Ensemble learning* (Ting and Witten, 1997; Zhou et al., 2002) is a technique that aims to improve the performance of a task by taking advantage of the complementarity of multiple systems. Generally, the different systems used in an ensemble result from different choices of architectures, training strategies, data ordering, and/or random initialization. In prompt ensembling (§6.1), the choice of prompt templates becomes another way to generate multiple results to be combined. This has the clear advantage that this does not necessarily require training the model multiple times. For example, when using discrete prompts, these prompts can simply be changed during the inference stage (Jiang et al., 2020c).

Few-shot Learning *Few-shot learning* aims to learn a machine learning system in the data-scarce scenarios with few training samples. There are a wide variety of methods to achieve few-shot learning including model agnostic meta-learning (Finn et al., 2017b) (learning features rapidly adaptable to new tasks), embedding learning (Bertinetto et al., 2016) (embedding each sample in a lower-dimensional space where similar samples are close together), memory-based learning (Kaiser et al., 2017) (representing each sample by a weighted average of contents from the memory) etc. (Wang et al., 2020). Prompt augmentation can be regarded as another way to achieve few-shot learning (a.k.a. priming-based few-shot learning (Kumar and Talukdar, 2021)). Compared to previous methods, prompt augmentation directly prepends several labeled samples to the currently-processed sample elicit knowledge from pre-trained LMs even without any parameter tuning.

Task	Example Prompt-Answer	Resource
Fact Probing	**Prompt** Adolphe Adam died in `[Z]`. **Answer** $\mathcal{V}$ **Prompt** iPod Touch is produced by `[Z]`. **Answer** $\mathcal{V}$ **Prompt** The official language of Mauritius is `[Z]`. **Answer** $\mathcal{V}$	LAMA dataset LPAQA dataset X-FACTR dataset
Text Classificatin	**Prompt** Which of these choices best describes the following document? "`[Class A]`", "`[Class B]`", "`[Class C]`". `[X][Z]` **Answer** `[Class A]`, `[Class B]`, `[Class C]` **Prompt** How is the text best described? : "`[Class A]`", "`[Class B]`" , or "`[Class C]`". `[X][Z]` **Answer** `[Class A]`, `[Class B]`, `[Class C]` **Prompt** This passage is about `[Z]`: `[X]` **Answer** `[Class A]`, `[Class B]`, `[Class C]` **Prompt** `[X]`. Is this review positive? `[Z]` **Answer** Yes, No **Prompt** `[X]` It was `[Z]`. **Answer** great, terrible	Meta [202]
Natural Language Inference	**Prompt** `[X1]`? `[Z]`, `[X2]` **Answer** Yes, No, Maybe **Prompt** `[X1]` `[Z]`, `[X2]` **Answer** Yes, No, Maybe	
Commonsense Reasoning	**Prompt** The trophy doesn't fit into the brown suitcase because `[Z]` is too large. **Answer** trophy, suitcase **Prompt** Ann asked Mary what time the library closes, because `[Z]` had forgotten. **Answer** Ann, Mary	PDP dataset WSC dataset CPRAG-102 dataset
Linguistic Knowledge Probing	**Prompt** A robin is a `[Z]`. **Answer** bird, tree **Prompt** A robin is not a `[Z]`. **Answer** bird, tree **Prompt** New is the opposite of `[Z]`. **Answer** old, young, current	WNLaMPro dataset ROLE-88 dataset NEG-136 dataset
Named Entity Recognition	**Prompt-Pos** `[X]` `[Span]` is a `[Z]` entity. **Prompt-Neg** `[X]` `[Span]` is not a named entity. **Answer** person, location, organization, miscellaneous **Prompt-Pos** The entity type of `Span` is `[Z]`. **Prompt-Neg** `[X]` The entity type of `[Span]` is none entity. Answer person, location, organization, miscellaneous	TemplateNER [29]
Question Answering	**Prompt** `[Question]` `[Passage]` `[Z]` **Prompt** `[Passage]` According to the passage, `[Question]` `[Z]` **Prompt** Based on the following passage, `[Question]` `[Z]`. `[Passage]`	
Summarization	**Prompt** Text: `[X]` Summary: `[Z]` **Prompt** `[X]` TL;DR: `[Z]` **Prompt** `[X]` In summary, `[Z]`	BARTScore [193]
Machine Translation	**Prompt** French: `[French sentence]` English: **Prompt** A French sentence is provided: `[French sentence]` The French translator translates the sentence into English: `[Z]` **Prompt** `[French sentence]` = `[Z]`	

Table 10: Commonly used prompts and answers for different tasks. `[X]` and `[Z]` denote slots for input and answer respectively. $\mathcal{V}$ denotes the vocabulary of the LM. More prompts for each task can be found using the **Resource** column.

Prompt Concept	Relevant Topic	Commonality	Peculiarity
Prompt Ensembling [68; 153]	Ensemble Learning [171; 204]	Combine results of multiple systems to get better performance	In prompt ensembling, multiple predictions result from different prompt variants. This contrasts with architecture or feature variations, each of which requires separate training.
Prompt Augmentation [16; 46]	Few-shot Learning [160; 42]	Use few examples to learn generalized rules	Prompt augmentation is a specific subset of few-shot learning.
	Larger-context Learning [18; 53]	Introduce larger context to aid the learning process	Additional information introduced in larger-context learning is not necessarily the labeled data.
Discrete Prompt Search [68; 159]	Query reformulation [123; 123]	Reformulate the input into a query form	Query reformulation commonly focuses on information extraction and question answering tasks, while prompt learning can be applied to a variety of NLP tasks
Discrete Prompt Fine-tuning [46]	QA-based multi-task learning [115; 97]	Reformulate many tasks into an QA form	QA-based formulations aim to solve different tasks through question answering, while prompting additionally targets full use of pre-trained models.
Continuous Prompt Fine-tuning [103; 36]	Controlled Text Generation [191; 77; 156]	Input is augmented with additional inputs to control the generation process	Controlled generation targets generation of a particular type of text while prompt learning uses prompts to specify the task itself.
Prompt-based downstream task learning [153; 193]	Supervised Attention [101; 165]	Require external hint to remind the model of which part information should be focused on	Research works on supervised attention usually target at salient information from an image or text, while prompt learning aims to utilize relevant knowledge from the pre-trained model.
	Data augmentation [40; 144]	Improving downstream tasks' performance by introducing additional samples	Data augmentation introduce additional training samples in an explicit way while prompts can be regarded as highly-condensed training samples [88].

Table 11: Other research topics relevant to prompting methods.

Larger-context Learning *Larger-context learning* aims to improve the system's performance by augmenting the input with additional contextual information, e.g. retrieved from the training set (Cao et al., 2018) or external data sources (Guu et al., 2020). Prompt augmentation can be regarded as adding relevant labeled samples into the input, but a minor difference is in larger-context learning, the introduced context is not necessarily labeled data.

Query Reformulation *Query reformulation* (Mathieu and Sabatier, 1986; Daumé III and Brill, 2004) is commonly used in information retrieval (Nogueira and Cho, 2017) and question answering tasks (Buck et al., 2017; Vakulenko et al., 2020), which aim to elicit more relevant texts (documents or answers) by expanding the input query with related query terms (Hassan, 2013) or generating paraphrases. There are several commonalities between prompt-based learning and query reformulation, for example (1) both aim to make better use of some existing knowledge bases by asking a right questions (2) the knowledge bases are usually a black-box, not available to the users, so researchers must learn how to probe it optimally based on solely questions.

There are also differences: the knowledge base in traditional query reformulation problems is usually a search engine (Nogueira and Cho, 2017), or QA system (Buck et al., 2017). By contrast, for prompt-based learning, we usually define this knowledge base as an LM, and need to find the appropriate query to elicit an appropriate answer from it. The input reformulation in prompt learning has changed the form of tasks. For example, an original text classification task has been converted into a cloze question problem, therefore bringing additional complexity regarding how to (1) make an appropriate task formulation, and (2) change the modeling framework accordingly. These steps are not required in traditional query formulation. Despite these discrepancies, some methodologies from query reformulation research still can be borrowed for prompt learning, such as decomposing input query into multiple sub-queries (Nogueira et al., 2019), similar to prompt decomposition.

QA-based Task Formulation *QA-based task formulation* aims to conceptualize different NLP tasks as a question-answering problem. (Kumar et al., 2016; McCann et al., 2018) are earlier works that attempt to unify multiple NLP tasks into a QA framework. Later, this idea has been further explored in information extraction (Li et al., 2020; Wu

et al., 2020) and text classification (Chai et al., 2020). These methods are very similar to the prompting methods introduced here in that they use textual questions to specify which task is to be performed. However, one of the key points of prompting methods is how to better use the knowledge in pre-trained LMs, and these were not covered extensively on previous works advocating for QA formulations.

Controlled Generation *Controlled generation* aims to incorporate various types of guidance beyond the input text into the generation model (Yu et al., 2020). Specifically, the guidance signals could be *style tokens* (Sennrich et al., 2016b; Fan et al., 2018), *length specifications* (Kikuchi et al., 2016), *domain tags* (Chu et al., 2017), or any variety of other pieces of information used to control of the generated text. It could also be *keywords* (Saito et al., 2020), *relation triples* (Zhu et al., 2020) or even *highlighted phrases or sentences* (Grangier and Auli, 2018; Liu et al., 2021c) to plan the content of generated texts. In a way, many of the prompting methods described here are a type of controllable generation, where the prompt is usually used to specify the *task itself*. Thus, it is relatively easy to find commonalities between the two genres: (1) both add extra information to the input text for better generation, and these additional signals are (often) learnable parameters. (2) If "controlled generation" is equipped with seq2seq-based pre-trained models (e.g., BART), then it is can be regarded as prompt learning with input-dependent prompts and the *prompt+LM fine-tuning* strategy (§7.2.5), e.g. *GSum* (Dou et al., 2021), where both the prompt's and pre-trained LM's parameters can be tuned.

Also, some clear discrepancies between controlled generation and prompt-based text generation are: (1) In controlled generation work, the control is generally performed over the style or content of the generations (Fan et al., 2018; Dou et al., 2021) while the underlying task remains the same. They don't necessarily require a pre-trained model. In contrast, the main motivation for using prompts for text generation is to specify the task itself and better utilize the pre-trained model. (2) Moreover, most of the current work on prompt learning in text generation shares a dataset- or task-level prompt (Li and Liang, 2021). Only very few works have explored input-dependent ones (Tsimpoukelli et al., 2021). However, this is a common setting and effective in the controlled text generation, which may provide valuable direction for the future work on prompt learning.

Supervised Attention Knowing to pay attention to the important information is a key step when extracting useful information from objects such as long text sequences (Liu et al., 2016; Sood et al., 2020), images (Sugano and Bulling, 2016; Zhang et al., 2020b), or knowledge bases (Yu et al., 2020; Dou et al., 2021)). *Supervised attention* (Liu et al., 2017b) aims to provide explicit supervision over the attention of models based on the fact that completely data-driven attention can overfit to some artifacts (Liu et al., 2017a). In this respect, prompt learning and supervised attention share ideas that both aim to extract salient information with some clues, which need to be provided separately. To solve this problem, supervised attention methods tried to use additional loss functions to learn to predict gold attention on a manually labeled corpus (Jiang et al., 2015; Qiao et al., 2018; Gan et al., 2017). Research on prompt learning may also borrow ideas from this literature.

Data Augmentation Data augmentation is a technique that targets increasing the amount of data that can be used for training by making modifications to existing data (Fadaee et al., 2017; Ratner et al., 2017). As recently observed by (Scao and Rush, 2021), adding prompts can achieve a similar accuracy improvement to the addition of 100s of data points on average across classification tasks, which suggests that using prompts for a downstream task is similar to conducting data augmentation implicitly.

10 Challenges

Although prompt-based learning has shown significant potential among different tasks and scenarios, several challenges remain, some of which we detail below.

10.1 Prompt Design

Tasks beyond Classification and Generation Most existing works about prompt-based learning revolve around either text classification or generation-based tasks. Applications to information extraction and text analysis tasks have been discussed less, largely because the design of prompts is less straightforward. We expect that applying prompting methods to these tasks in the future it will require either reformulating these tasks so that they can be solved using classification or text generation-based methods, or performing effective answer engineering that expresses structured outputs in an appropriate textual format.

Prompting with Structured Information In many NLP tasks, the inputs are imbued with some variety of structure, such as tree, graph, table, or relational structures. How to best express these structures in prompt or answer engineering is a major challenge. Existing works (Chen et al., 2021b) make a step by making prompts with additional marks to encode lexical information, such as entity markings. Aghajanyan et al. (2021) present structured prompts based on hyper text markup language for more fine-grained web text generation. However, moving beyond this to more complicated varieties of structure is largely unexplored, and a potentially interesting research area.