Position-based Prompting for Health Outcome Generation

Micheal Abaho[1] **Danushka Bollegala**[1,2*] **Paula R Williamson**[1] **Susanna Dodd**[1]
[1]University of Liverpool, United Kingdom
[2]Amazon
{m.abaho,danushka,prw,shinds}@liverpool.ac.uk

Abstract

Probing factual knowledge in Pre-trained Language Models (PLMs) using prompts has indirectly implied that language models (LMs) can be treated as knowledge bases. To this end, this phenomena has been effective, especially when these LMs are fine-tuned towards not just data, but also to the style or linguistic pattern of the prompts themselves. We observe that, satisfying a particular linguistic pattern in prompts is an unsustainable, time-consuming constraint in the probing task, especially because, they are often manually designed and the range of possible prompt template patterns can vary depending on the prompting task. To alleviate this constraint, we propose using a position-attention mechanism to capture positional information of each word in a prompt relative to the mask to be filled, hence avoiding the need to re-construct prompts when the prompts' linguistic pattern changes. Using our approach, we demonstrate the ability of eliciting answers (in a case study on health outcome generation) to not only common prompt templates like Cloze and Prefix, but also rare ones too, such as Postfix and Mixed patterns whose masks are respectively at the start and in multiple random places of the prompt. More so, using various biomedical PLMs, our approach consistently outperforms a baseline in which the default PLMs representation is used to predict masked tokens.

1 Introduction

Language models (LMs) as knowledge bases (KBs) (LM-as-KB) is a rapidly growing phenomenon attracting a lot of attention in the Natural Language Processing (NLP) community (Petroni et al., 2019; Brown et al., 2020; Shin et al., 2020; Schick and Schütze, 2020b). LM-as-KB implies the usage

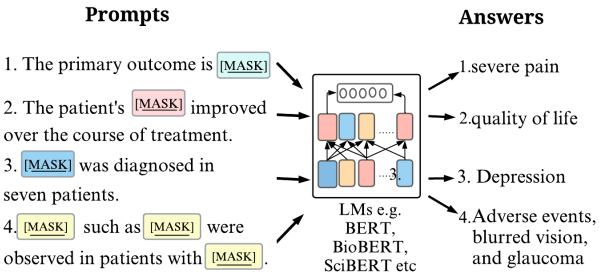

Figure 1: Prompt query variants used for probing evidence (in form of health outcomes) from PLMs, including common styles like Prefix (1) and Cloze (2) style, as well as rare styles Postfix (3) and Mixed (4) styles with [MASK] token/s at the beginning and in multiple positions in the prompt.

of LMs as an alternative or at least a proxy for explicit KBs. To achieve LM-as-KB, researchers adopt prompt-based learning (PBL) in which LMs learn to probabilistically predict missing information once given fill-in-the-blank prompt inputs (Liu et al., 2021) such as "Eiffel tower is located in ___". PBL has generally been a success, for example, in a systematic survey of prompting methods, Liu et al. (2021) indicate that *"pre-train, prompt and predict"* is a new paradigm replacing *"pre-train and fine-tune"* paradigm in NLP. Because of this success, the rationale that LMs contain factual retrievable knowledge (LM-as-KB) is ostensibly justified and therefore continually explored.

The prompt sequences often used in PBL have a masked token or span (denoted by [MASK] in the remainder of the paper) that positionally appears either in the middle (Cloze-style) (Petroni et al., 2019; Schick and Schütze, 2020b; Cui et al., 2021) or at the very end of the sequence (Prefix style) (Qin and Eisner, 2021; Shin et al., 2020). Moreover, we learn that the majority of the PBL tasks probe relational knowledge possessed by pre-trained language models (PLMs) (Jiang et al., 2020b; Petroni et al., 2019; Davison et al., 2019), which implies that the prompt inputs used in querying the PLMs have to contain relational information (such as

*Danushka Bollegala holds concurrent appointments as a Professor at University of Liverpool and as an Amazon Scholar. This paper describes work performed at the University of Liverpool and is not associated with Amazon.

"subject-relation-object" triples). Furthermore, we observe that, a fair amount of time in several PBL tasks is spent reconstructing prompt inputs through manually designing templates (Petroni et al., 2019; Davison et al., 2019) or corrupting prompt inputs through deletion (Lewis et al., 2019), replacement (Raffel et al., 2019) or permutation (Heinzerling and Inui, 2020).

As discussed above, we notice that, the syntactic and semantic structure of prompt inputs is a constraint encountered in PBL, notwithstanding the multitude of constraints that could arise given that PBL is inherently a text generation task (Liu et al., 2021). This constraint will usually require researchers to laboriously prepare supervised data with prompts whose linguistic patterns suit the objective of the prompting task, For instance, (Davison et al., 2019; Jiang et al., 2020a; Heinzerling and Inui, 2020), use templates that reformulate prompts to contain relational information connecting a particular text span to the to-be filled information. However, template-based prompt reformulation has two main challenges. First, it presents a risk of corrupting the grammar of the prompts unwittingly (Davison et al., 2019). Second, the search space of the candidate prompts is too large (Gao et al., 2020) and is practically impossible to create templates that can enumerate all possible linguistic patterns that prompt queries can be tailored to. For example, prompt template patterns with missing information at the beginning and or with multiple missing information in a sequence are yet to be explored in prior works.

To address the above-mentioned challenges, we propose a strategy we denote position-based prompting (PBP), which is less concerned about the linguistic pattern or shape the prompt takes on, but rather focuses on the words (that the prompts are composed of) and their positions relative to the [MASK]. PBP is focused on shifting the emphasis on subject-relation-object triples to the masked positions as well as the interaction of all the other words with the [MASK]s position. PBP is built to automatically adjust from one prompt template to another, which essentially eliminates the need to prepare hand crafted prompts in the event that an LM is to be probed for rare knowledge. In its architecture, PBP enhances contextualised word representations with position-aware representations to solve fill-in-the-blank tasks. In our approach, we fine-tune PLM parameters along with position-oriented parameters to generate position-based contextualised word representations.

To test our approach, we investigate how well biomedical LMs store and recall information relevant to biomedical entities, with a specific interest in health outcomes, which are defined as measurements or observations used to capture and assess the effect of treatments (Williamson et al., 2017). In addition to the Prefix and Cloze styles, we incorporate two rare prompt style patterns that we denote Postfix and Mixed, where the former contains the [MASK] token/s at the beginning of the prompt sequence and the latter has multiple [MASK] token/s in various positions (Figure 1). Our approach obtains mean scores (across several biomedical LMs) in Exact Match (EM) and Partial Match (PM) metrics that are an improvement (2.4% across both metrics) over those obtained using the vanilla PLM representations, reporting a significant improvement of 6.49% in F1 on the EBM-NLP (Nye et al., 2018) dataset. As later defined in section 4.1, EM measures the percentage of predictions of all [MASK] tokens (or spans) that match the ground truth, whereas PM measures the percentage of correctly predicted [MASK] tokens.

2 Entity memorisation and recalling

Large-scale LMs with billions of parameters have already shown to recall facts that were observed in the training data (Heinzerling and Inui, 2020; Jiang et al., 2020a). However, the ground truth for these LMs to achieve this is already laid with systematically handcrafting rules to follow in creating the prompt input sequences they receive at the training stage. For instance, the majority of the prompts created in PBL tasks embed knowledge in form of triples {*subject,relation,object*} such that LMs could correctly predict *object* entities when prompted with a sequence containing a *subject* and *relation* or otherwise predict *subject* entities when prompted with a sequence containing an *object* and a *relation* (Sung et al., 2021; Jiang et al., 2020a; Qin and Eisner, 2021). Whichever the case, models often predict answers as shown in (1).

$$\hat{y}_i = \underset{y_i}{\operatorname{argmax}} \; p([\text{MASK}] = y_i | x_{prompt}) \quad (1)$$

where i is the position of masked token within a prompt x_{prompt}.

In this work, we however do not assume any prior knowledge contained in a prompt, but rather

simply locate outcome entities in the sentences extracted from Randomised Clinical Trial (RCT) abstracts and mask them, an approach we refer to as *custom masking*.

3 Method

In addition to formally defining the task we undertake, this section discusses the data used as well as the different stages of our proposed PBP strategy.

3.1 Task

Let us consider an input prompt sequence s with one or more outcomes masked such that $s = x_1, \ldots [M]_i \ldots [M]_j \ldots x_n$, where $[M]$ is a masked token sequence, $[M] = \{x_i\}_{i\geq 1}^{i+|M|}$, $i \in [1, n]$ and $|M|$ is the length of the masked sequence. We consider four different prompt query variants shown in Figure 1: **Prefix prompts** contain $[M]$ at the end of the prompt, **Cloze prompts** contains $[M]$ in the middle of the prompt, **Postfix prompts** contain $[M]$ at the start of the prompt, and **Mixed prompts** where there are several masked sequences distributed across the prompt. The questions we then pose are: (a) *can we determine how knowledgeable biomedical PLMs are of stored facts such as health outcomes?*, and (b) *If queried with any of the above variants, would these PLMs correctly fill in $[M]$s with the correct outcomes?*

3.2 Datasets

Different from previous PBL works, we neither create custom templates nor do we reformulate prompts to follow an ideal linguistic pattern. We use plain raw sentences (that mention health outcomes) extracted from RCT PubMed abstracts, which are contained in the revised version of EBM-NLP (Abaho et al., 2019) and EBM-COMET (Abaho et al., 2021b) datasets. Both of these datasets support evidence based medicine (EBM) tasks such as extraction of health outcomes from clinical trials (Beltagy et al., 2019; Abaho et al., 2021a).

We do not eliminate any of the abstract sentences that do not mention outcomes, because we aim to familiarise the PLM (at fine-tuning) with text or context in RCT abstracts which generally report about outcomes during clinical trial studies (Williamson et al., 2017). We refer to these sentences as *no_blank sequences* and use them alongside the prompt query variants introduced earlier. To our advantage, several sentence segments have no outcome annotations in both the EBM-NLP and EBM-COMET datasets.

3.3 Masked Language model and Prompt engineering

We extract a hidden state h_i for each token in an input prompt s using a domain-specific PLM,

$$\boldsymbol{h_i} = \text{PLM}_\theta(x_i) \tag{2}$$

where $\boldsymbol{h_i}$ is a hidden state for the word x at position i. The matrix of hidden states for the entire input prompt is represented as $\mathbf{H} \in \mathbb{R}^{n\times k}$, where n is number of words in s and k is the hidden state size.

We define a function $f_{\textbf{prompt}}$ that concatenates the h_i in (2) to a randomly initialised d dimensional vector, which we denote as z_t corresponding to one of the four prompt query variants or the additional *no_blank sequences* (introduced in §3.2), where $t \in$ [*prefix, cloze, postfix, mixed, no_blank*]. The function ensures that if an input s is a Prefix prompt, the corresponding vector z_{prefix} is concatenated to each h_i generated from s as shown in (3). This is done to enable knowledge transfer from one prompt query to another. For example, Mixed prompts are by construction a combination of Prefix, Postfix, and Cloze, hence they should benefit from information sharing via a common vector space.

$$f_{\textbf{prompt}}(h_i) = [z_t; h_i] \tag{3}$$

$z_t \in \mathbb{R}^{d_t}$, where z_t is a query type embedding of size d_t.

3.4 Position based conditioning (PBC)

To enrich the token representations, we propose a position-based attention mechanism to steer the model's focus on relevant information in the input prompt. We define a sequence of position ids for each input prompt, where all masked positions take on an id of 0 and all the other tokens take id's relative to the masked position id. For example given a Cloze prompt with m tokens, we assign a mask at position i an id 0, and resulting sequence of position ids is $p = [1-i, 2-i, \ldots, -1, 0, 1, \ldots, (m-1)-i, m-i]$. We compute an attention vector $A^{(s)}$, given by (4), for an input prompt s that allows each token to interact with every other token and retain knowledge of the relative position of the masked tokens in the input sequence.

$$\boldsymbol{A^{(s)}} = \text{softmax}(\mathbf{V}^\top \tanh(\mathbf{W}\mathbf{H}^\top + \mathbf{U}\mathbf{P}_s^\top)) \tag{4}$$

Here, $\boldsymbol{A}^{(s)} \in \mathbb{R}^{n\times 1}$, $\mathbf{V} \in \mathbb{R}^{k_a\times 1}$, k_a is size of attention layer, $\mathbf{W} \in \mathbb{R}^{k_a\times k}$, $\mathbf{P}_s \in \mathbb{R}^{n\times k_p}$ and $\mathbf{U} \in \mathbb{R}^{k_a\times k_p}$. $\mathbf{P}_s$ is a matrix of position embeddings of size k_p extracted for each position p_n in the input prompt s. These embeddings are extracted from a trainable matrix $\mathbf{P} \in \mathbb{R}^{2n\times k_p}$ of randomly initialised vectors of size k_p for all possible positions $2n$ where n is the maximum sequence length, $|\{p_n\}_{-n}^{n-1}| = 2n$. The position based representation of each token is then computed with respect to the type of prompt. For the Prefix, Postfix and Cloze prompts, we obtain a prompt representation M^s given by (5).

$$\mathbf{M}^{(s)} = \boldsymbol{A}^{(s)}\mathbf{H} \tag{5}$$

Here, $\mathbf{M}^{(s)} \in \mathbb{R}^{n\times k}$. For the Mixed prompts in which we have multiple masked positions within the input sequence, we avoid biasing the attention mechanism towards masks at a specific position and thereby considering as many position id sequences as there are masked positions in the input prompt. For example, given a sequence with 3 masked positions, $s = [M], x_2, x_3, [M], x_5, x_6, [M]$, we obtain 3 position id sequences, i.e. the combined position id sequences is,

$$P^{(s)} = \bigcup_i P_i,$$

where each P_i is obtained with respect to the current mask position i. For the example above, we have $P^{(s)}$ = {[0,1,2,3,4,5,6], [-3,-2,-1,0,1,2,3,], [-6,-5,-4,-3,-2,-1,0]}, where the first position id sequence is obtained by treating the $[M]$ at position 1, as mask at i, the second is obtained by treating the $[M]$ at position 4 as mask at i and finally the third by treating $[M]$ at the last position as mask at i. Attention vectors are computed for each position id sequence (P_i) and subsequently used to obtain the prompt representation $\mathbf{M}^s_{P_i}$. We compute the final representation of a Mixed prompt as the mean pool across these different representations,

$$\mathbf{M}^{(s)} = \sum_i^{|P^{(s)}|} \mathbf{M}^s_{P_i} \tag{6}$$

3.5 Prompt fine-tuning

The predicted probability of each vocabulary token is estimated via (7).

$$y = \text{softmax}(f(W_v{\mathbf{M}^{(s)}}^\top) \tag{7}$$

Therein, $W_v \in \mathbb{R}^{v^*\times k}$, v^* is the vocabulary size and f is a non-linear activation function. We use a BERT-based loss in predicting the masked tokens in each input given by (8).

$$\boldsymbol{L}_{PLM} = -\sum_{s\in\mathcal{T}}\sum_i^n \log P(y_i|s) \tag{8}$$

where $\mathcal{T}$ is the set of training example prompts. Some of the prompt query variants (Postfix and Prefix) are rare in the datasets, and some other prompt sequences are quite lengthy. This poses a challenge particularly when using small PLMs (with few parameters) to recall factual information. In order to mitigate model forgetfulness in such examples, we introduce an auxiliary task that computes a text classification loss as a cross entropy loss given by (9).

$$\boldsymbol{L}_{TC} = -\sum_{s\in\mathcal{T}}\sum_{i\in n} \log P(y_i|y_{<i}, s) \tag{9}$$

The overall training loss is defined as the weighted combination of the two losses as given in (10).

$$\boldsymbol{L} = \boldsymbol{L}_{PLM} + \lambda\boldsymbol{L}_{TC} \tag{10}$$

Similar to (Chronopoulou et al., 2019) and (Schick and Schütze, 2020a), we introduce a weighting parameter $\lambda(> 0)$ to adapt the auxiliary losses to the main mask prediction task.

3.6 Prediction

Similar to BERT (Devlin et al., 2018), we consider generating outputs in parallel, initially treating the default representations provided by the model in (2) as a baseline and therefore use them to predict tokens in masked positions. We then use position-aware representation obtained using the attention mechanism in §3.4 to predict the mask tokens, calling these results Position-based conditioning (PBC). Lastly, we endeavour to retain the contextual knowledge presented by the PLMs as much as we possibly can by computing an average of the Baseline and PBC representations and term these Contextual PBC.

4 Experiments

In our experiments, we use several PLMs that are pre-trained on clinical texts such as PubMed abstracts, which often report outcomes such as BioBERT (Lee et al., 2020), SciBERT (Beltagy

Dataset-	EBM-COMET						EBM-NLP					
Method-	Baseline		PBC		Contextual PBC		Baseline		PBC		Contextual PBC	
Metric-	EM	PM	EM	PM	EM	PM	EM	PM	EM	PM	EM	PM
BERT	43.12	47.55	43.04	49.84	44.32	55.94	37.40	45.55	41.10	47.00	47.31	51.06
BioBERT	50.71	58.01	50.55	58.61	53.34	59.65	51.15	55.62	51.19	53.80	52.15	54.50
SciBERT	61.17	67.48	62.34	69.85	63.00	70.95	57.12	62.25	57.18	63.75	59.44	63.91
Biomed_RoBERTa	44.01	59.67	44.32	59.73	44.32	62.86	40.45	51.72	47.21	49.81	49.17	55.00
UmlsBERT	31.05	34.61	30.47	35.77	31.88	36.46	28.66	33.15	30.02	38.51	39.16	40.15
Mean score	46.01	53.46	46.14	54.76	47.37	57.17	42.96	49.66	45.34	50.57	49.45	52.92

Table 1: Table reports EM and PM accuracies of the various biomedical Pre-trained Language Models for the outcome recalling experiments. Mean score in a particular column is the average across all results in that column.

et al., 2019) and Biomed_RoBERTa (Gururangan et al., 2020). Additionally, we include UmlsBERT because it augments BERT's pre-training input with semantic type embeddings aligned to clinical knowledge (semantic types) in the Unified Medical Language System (UMLS) Metathesaurus (Michalopoulos et al., 2020). We also use BERT (Devlin et al., 2018) as a vanilla PLM that has not been pre-trained specifically on clinical texts.

4.1 Training and Evaluation

Unlike previous works where a particular relation within a prompt e.g. born-in, lives-in etc. might appear multiple times within the train set, in our case, prompts are not semantically related in any way (i.e. their is no relation knowledge that can be transferred over from one prompt to another). Because of the nature of our prompts, we believe it might be harder for the model to memorise them, we therefore opt to train the models until the perplexity on the training data reaches 1 or until the accuracy on the validation data saturates. We examine the model's generalisation ability to transfer knowledge to unseen prompts in few-shot and zero-shot settings. For the few-shot setting, we design experiments where we measure a model's accuracy in generating outcomes (as answers), which it encountered in a small number of prompts during training. The contexts in these evaluation prompts are not encountered during training. For example, consider an evaluation prompt – *"The patient's overall [MASK] improved according to the HRQOL questionnaire*", the model would not have encountered the context surrounding the *"[MASK]"*. For the zero-shot evaluation, the model would have neither encountered the prompt nor the target outcomes during training. To simulate both the zero- and few-shot settings, we randomly split the datasets into train (80%) and test (20%) splits, and use the latter for the generalisation evaluation task shown in Table 3. We tune all hyperparameters using the validation data, and obtain optimal values as follows: learning rate - 5e-5, batch size - 8, query type embedding size - 50, position embedding size - 300 and an attention layer size - 200. Further details on tuning bounds are provided in the Appendix.

Metrics: We define two different metrics for evaluating the proposed PBP strategy: Exact Match (EM) and Partial Match (PM). EM counts a prediction as 1 only if it matches completely with the correct answer, whereas PM uses the fraction of the overlapping tokens between the predicted and correct answers. Both EM and PM are averaged over all test instances to compute aggregated evaluation metrics, and we report their percentages in the paper.

5 Results

In this section, we evaluate how well the model generates health outcomes when queried to answer a given prompt. For example, "*After patients were given sorafenib, they reported [MASK]*", the model should correctly generate the outcome *Fatigue* for the *[MASK]*.

5.1 Outcome memorisation and retrieval

Table 1 shows the performance of the proposed PBC method in the outcome generation task. As observed, PBC consistently outperforms the baseline across most of the clinically informed BERT LMs (for both datasets), particularly for the PM results. More interestingly, we notice that Contextual PBC further improves the performance (both in EM and PM), indicating the importance of preserving the contexts in the position-based representations.

Comparing the different LMs, we found that, SciBERT performs best followed by

	#	Average prompt length	EM	PM
Postfix	65	18.5	48.43	58.51
Prefix	53	9.1	69.23	77.24
Cloze	630	24.2	50.08	60.49
Mixed	2594	38.8	43.68	45.46

Table 2: Exact Match (EM) and Partial Match (PM) accuracies for Outcome memorisation/recalling for the different prompt types using the EBM-COMET dataset.

	Cloze	Mix	Postfix	Prefix
#	174	613	13	12

Table 3: Number of prompts per prompt type used in evaluation of the few- and zero-shot settings.

Biomed_RoBERTA and BioBERT. Since all tested models follow the original BERT's architecture, we hypothesize that, the nature of corpora used in pre-training the best performing models was responsible for the performance, i.e. unlike UMLSbert and BERT, all the other models are pre-trained on text that includes PubMed abstracts, which often report outcomes. Additionally, we observe that PM results were generally better than EM results, which we attribute to the fact that PM is less strict compared to EM because it rewards the model for correctly generating a few of the tokens in the masked positions. Overall, the results suggest that PBC can be used to effectively retrieve facts such as health outcomes (biomedical entities) by simply augmenting contextual word representations with position-aware representations.

5.1.1 Prompt query variants

In Table 2, we notice that the accuracy with which a model correctly answers Prefix prompts is significantly higher than that of the other prompts. We attribute this performance to the short length of these spans such as the one shown in Table 4 and the average number of tokens to decode per prompt. We also notice that the model struggles to correctly answer Mixed prompts compared to other types of prompts. We attribute this to the fact that, Mixed prompts are generally very long sequences (38.8 tokens on average) and contain multiple masked positions to be predicted.

5.2 Few- and Zero-shot Evaluations

To evaluate the model's generalisability, we fine-tune the model towards a small amount of target

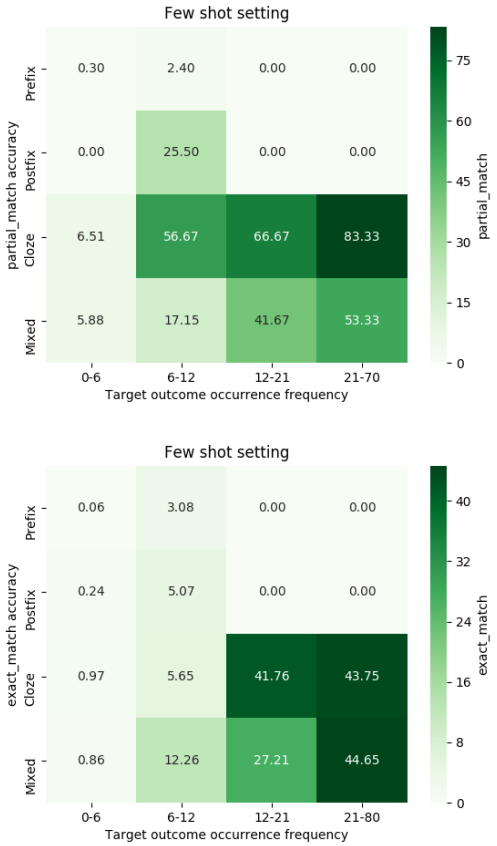

Figure 2: Visualizing the Partial Match and Exact match accuracies when the best model (SciBERT+Contextual PBC+EBM-COMET) is trained with only a certain number of target outcomes.

outcomes, and then measure the transferability of this knowledge by requiring the model to accurately generate these outcomes in prompts with completely different contexts. Test set prompts in Table 3 are carefully chosen using regular expression matching such that the contexts surrounding the missing outcomes are different from that of similar outcomes observed during training. For example, the model could have been trained on the outcome "adverse events" in five different prompts, and then at evaluation, the model is required to generate the same outcome, however using prompts that are different from those encountered during training. By *different* here we mean that the context (e.g. {ctxt} surrounding masks [M] in Table 4) in the prompt changes during this evaluation. Figure 2 plots shows results of model evaluation on prompts (Table 3). As observed in the plots, the model struggles to generate outcomes it hardly encountered during training (i.e. outcomes appearing in 0-6 prompts or 6-12 prompts). This is mostly evident in generating outcomes for Prefix and Postfix

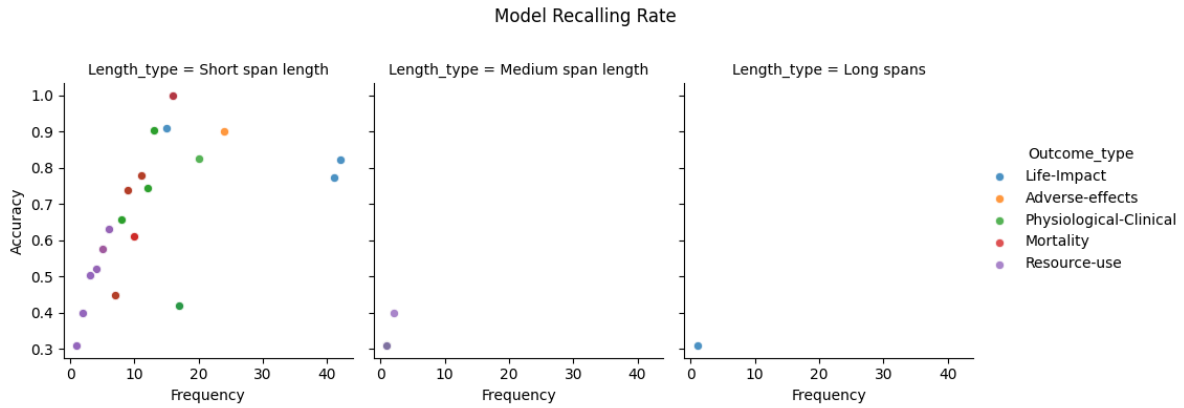

Figure 3: Analysis of the accuracy (PM) with which best model (SciBERT+Contextual PBC+EBM-COMET) recalls different types of factual information (outcome types) with varying span lengths and occurrence frequency (in the dataset).

prompts, which is because there were not just few evaluated prompts of this types, but there were also few (53 and 65 respectively as shown in Table 2) in the train set. However, we see a trend of performance improvement when the frequency of target outcomes encountered during training increases, particularly for the Mixed and Cloze prompt.

6 Analysis

6.1 Impact of Length and Frequency of Outcomes

We partition the entire set of outcomes in EBM-COMET into 3 different groups based on lengths. Dividing the length of the longest outcome (22) by 3, we get approximately 7 which we use to create 3 groups i.e. 1) "short span length" to represent outcomes that are ≤ 7 tokens long, 2) "medium span length" to represent outcomes of $7 >$ and ≤ 14 tokens, and finally 3) "long spans" to represent outcomes of ≥ 14 tokens long. Figure 3 shows how well the best model (SciBERT+Contextual PBC+EBM-COMET) performs when recalling outcomes of varying lengths and frequencies. Following prior work on EBM NLP, we endeavour to show the model's outcome recall rate by outcome type, which can be informative in terms of the complexity of modelling these outcomes. We firstly notice the skewed distribution of outcome lengths with short spans dominant in the training sample. Unsurprisingly, we observe a trend of a performance increase as the frequency increases across the left hand plot with short outcomes, implying that the model struggles to recall infrequent outcomes despite their size but easily recalls the more frequent ones.

6.2 Random masking Vs custom masking

Figure 4 shows results of an ablation test in which we replace our custom masking approach with random masking. The key difference between the two is, while custom masking involves masking (or hiding) the outcomes in the prompts, random masking arbitrary masks 15% of the prompts tokens. As shown in the figure, the number of epochs required to reach a perplexity of 1.0 on the train data for the two masking approaches is almost incomparable, with custom masking quickly achieving this in approximately 7 epochs and random masking failing to achieve this, even after 20 epochs. The earliest random masking achieves 1.0 perplexity is 80 epochs for SciBERT, however we only visualise 20 epochs because of space. Besides this, the insight suggests that, custom masking would significantly reduce GPU run-time or otherwise minimise overwhelming computational resources with massive datasets.

6.3 Error Analysis

We analyse the outcomes generated by the best model (SciBERT+Contextual PBC+EBM-COMET) during the few shot evaluation and notice that whilst the model generates correct outcomes for some prompts, it makes various kinds of mistakes. Table 4 includes a fair sample of the most commonly discovered mistakes. **Incomplete outcomes**, such in the Postfix where instead of "Quality of life", the model generates "Life". **Outcomes with irrelevant information**, such as Prefix case where the models generates more than what's expected, "unwanted pain" instead of "pain". Finally, **wrong outcomes**, where the model generates completely unexpected outcomes such as the case in

Query Variant	Prompt	Correct	Generated outcomes
Cloze {ctxt} [M] {ctxt}	Self-reported life-time medical diagnosis of **[M]** or use of antidepressants was considered as outcome.	- Depression	- Depression
Postfix [M] {ctxt}	**[M]** was assessed by questionnaires EORTC QLQ-C30, and EORTC QLQ-BR23 at baseline, and at three, six, and nine months.	- Quality of life	- Life
Prefix {ctxt} [M]	Two CMZ patients and one morphine patient showed complete **[M]**.	- pain	- unwanted pain
Mixed {ctxt} [M] {ctxt} [M] {ctxt}	Further additional benefits are better **[M]** and shorter **[M]** compared with standard GVHD prophlaxis without ATLG.	- quality of life (QOL) - immunosuppressive treatment	- immunosuppressive treatment
	The incidence of postoperative **[M]**, **[M]**, **[M]** and **[M]** was similar between the groups	- nausea, - vomiting, - drowsiness, -headache	- anxiety, - depression

Table 4: Example prompts from our test set and their predicted or generated outcomes for the outcome generation task. The Query variant column indicates the type of prompt as well as the prompt structure where {ctxt} implies context which might appear before, after or either ends of a masked sequence span.

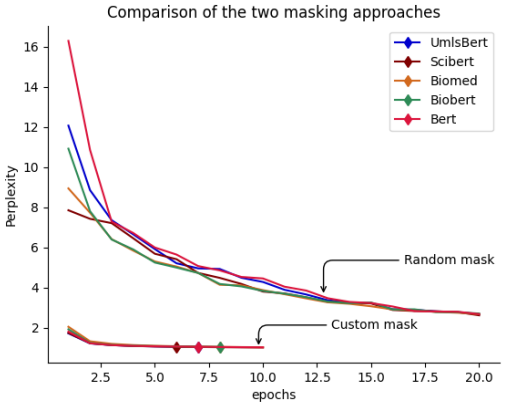

Figure 4: Achieving a target perplexity of 1.0 on the train dataset takes no fewer than 20 epochs with generic random masking of 15% of the input prompt tokens (Devlin et al., 2018) compared to masking target factual information i.e. outcome spans themselves. Hitting target perplexity is shown using a diamond.

.

the Mixed prompts.

7 Related work

Interrogating PLMs with fill-in-the-blank prompts to determine their knowledge and awareness of factual information is a trending paradigm in NLP. Despite the emergence of subtle techniques such as automating prompt structuring (Shin et al., 2020; Gao et al., 2020), selectively updating parameters of LMs and prompts (also known as continuous prompting) (Li and Liang, 2021; Qin and Eisner, 2021), or even not tuning at all (Brown et al., 2020), several works including these still heavily rely on handcrafted prompts to use in probing LMs. Our efforts are motivated by the fact that we need not worry about the nature of the prompt, but rather can leverage on information local to the prompt such as word positions to probe the LMs. We attempt to enhance a word's contextualised representation with position based representations to capture the word's position relative to the mask to be filled. Previously some works have used similar position-aware attention over LSTMs for relation extraction, sequence labelling and slot filling tasks in different datasets (Wei et al., 2021; Zhang et al., 2017). To the best of our knowledge, we are the first to use an extra position-attention layer above transformer models such as BERT to solve the fill-in-the-blank prompting task.

8 Conclusion

This paper assesses the possibility of ignoring the constraint of aligning prompts to specific linguistic patterns in prompting tasks that aim to store knowledge in LMs that could later be retrieved or transferred for fact generation tasks. In experiments using clinical domain datasets (supporting EBM tasks), we show that the position-based attention implemented over contextualised LMs can improve the ability of PLMs to recall facts such as outcomes (biomedical entities) encountered during training. We further observe our proposed model is able to generalise across unseen prompts, performing considerably well for Cloze and Mixed (extremely rare in PBL tasks) prompts. With the obtained experimental results, despite not aligning our prompts to commonly followed linguistic patterns, we can positively answer the question posed in §3.1 by claiming that PLMs are knowledgeable of stored facts.

References

Michael Abaho, Danushka Bollegala, Paula Williamson, and Susanna Dodd. 2021a. Detect and classify–joint span detection and classification for health outcomes. *arXiv preprint arXiv:2104.07789*.

Micheal Abaho, Danushka Bollegala, Paula Williamson, and Susanna Dodd. 2019. Correcting crowdsourced annotations to improve detection of outcome types in evidence based medicine. In *CEUR Workshop Proceedings*, volume 2429, pages 1–5.

Micheal Abaho, Danushka Bollegala, Paula Williamson, and Susanna Dodd. 2021b. Assessment of contextualised representations in detecting outcome phrases in clinical trials. *European Journal of Biomedical Informatics*, 17(9).

Iz Beltagy, Kyle Lo, and Arman Cohan. 2019. Scibert: A pretrained language model for scientific text. *arXiv preprint arXiv:1903.10676*.

Austin J Brockmeier, Meizhi Ju, Piotr Przybyła, and Sophia Ananiadou. 2019. Improving reference prioritisation with pico recognition. *BMC medical informatics and decision making*, 19(1):1–14.

Tom B Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *arXiv preprint arXiv:2005.14165*.

Alexandra Chronopoulou, Christos Baziotis, and Alexandros Potamianos. 2019. An embarrassingly simple approach for transfer learning from pretrained language models. *arXiv preprint arXiv:1902.10547*.

Leyang Cui, Yu Wu, Jian Liu, Sen Yang, and Yue Zhang. 2021. Template-based named entity recognition using bart. *arXiv preprint arXiv:2106.01760*.

Joe Davison, Joshua Feldman, and Alexander M Rush. 2019. Commonsense knowledge mining from pretrained models. 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 1173–1178.

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*.

Susanna Dodd, Mike Clarke, Lorne Becker, Chris Mavergames, Rebecca Fish, and Paula R. Williamson. 2018. A taxonomy has been developed for outcomes in medical research to help improve knowledge discovery. *Journal of Clinical Epidemiology*, 96:84–92.

Tianyu Gao, Adam Fisch, and Danqi Chen. 2020. Making pre-trained language models better few-shot learners. *arXiv preprint arXiv:2012.15723*.

Suchin Gururangan, Ana Marasović, Swabha Swayamdipta, Kyle Lo, Iz Beltagy, Doug Downey, and Noah A. Smith. 2020. Don't stop pretraining: Adapt language models to domains and tasks. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8342–8360, Online. Association for Computational Linguistics.

Benjamin Heinzerling and Kentaro Inui. 2020. Language models as knowledge bases: On entity representations, storage capacity, and paraphrased queries. *arXiv preprint arXiv:2008.09036*.

Xiaoli Huang, Jimmy Lin, and Dina Demner-Fushman. 2006. Evaluation of PICO as a knowledge representation for clinical questions. *AMIA ... Annual Symposium proceedings. AMIA Symposium*, pages 359–63.

Zhengbao Jiang, Antonios Anastasopoulos, Jun Araki, Haibo Ding, and Graham Neubig. 2020a. X-factr: Multilingual factual knowledge retrieval from pretrained language models. *arXiv preprint arXiv:2010.06189*.

Zhengbao Jiang, Frank F Xu, Jun Araki, and Graham Neubig. 2020b. How can we know what language models know? *Transactions of the Association for Computational Linguistics*, 8:423–438.

Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. 2020. Biobert: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, 36(4):1234–1240.

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*.

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

Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig. 2021. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *arXiv preprint arXiv:2107.13586*.

George Michalopoulos, Yuanxin Wang, Hussam Kaka, Helen Chen, and Alex Wong. 2020. Umls-bert: Clinical domain knowledge augmentation of contextual embeddings using the unified medical language system metathesaurus. *arXiv preprint arXiv:2010.10391*.

Benjamin Nye, Junyi Jessy Li, Roma Patel, Yinfei Yang, Iain J. Marshall, Ani Nenkova, and Byron C. Wallace. 2018. A Corpus with Multi-Level Annotations of Patients, Interventions and Outcomes to Support Language Processing for Medical Literature. In *Proc.*

of the 56th Annual Meeting of the Association for Computational Linguistics, pages 197–207.

Fabio Petroni, Tim Rocktäschel, Sebastian Riedel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, and Alexander Miller. 2019. Language models as knowledge bases? 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 2463–2473, Hong Kong, China. Association for Computational Linguistics.

Guanghui Qin and Jason Eisner. 2021. Learning how to ask: Querying lms with mixtures of soft prompts. *arXiv preprint arXiv:2104.06599*.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2019. Exploring the limits of transfer learning with a unified text-to-text transformer. *arXiv preprint arXiv:1910.10683*.

Timo Schick and Hinrich Schütze. 2020a. Exploiting cloze questions for few shot text classification and natural language inference. *arXiv preprint arXiv:2001.07676*.

Timo Schick and Hinrich Schütze. 2020b. It's not just size that matters: Small language models are also few-shot learners. *arXiv preprint arXiv:2009.07118*.

Taylor Shin, Yasaman Razeghi, Robert L Logan IV, Eric Wallace, and Sameer Singh. 2020. Autoprompt: Eliciting knowledge from language models with automatically generated prompts. *arXiv preprint arXiv:2010.15980*.

Mujeen Sung, Jinhyuk Lee, Sean Yi, Minji Jeon, Sungdong Kim, and Jaewoo Kang. 2021. Can language models be biomedical knowledge bases? *arXiv preprint arXiv:2109.07154*.

Wei Wei, Zanbo Wang, Xianling Mao, Guangyou Zhou, Pan Zhou, and Sheng Jiang. 2021. Position-aware self-attention based neural sequence labeling. *Pattern Recognition*, 110:107636.

Paula R. Williamson, Douglas G. Altman, Heather Bagley, Karen L. Barnes, Jane M. Blazeby, Sara T. Brookes, Mike Clarke, Elizabeth Gargon, Sarah Gorst, Nicola Harman, Jamie J. Kirkham, Angus McNair, Cecilia A.C. Prinsen, Jochen Schmitt, Caroline B. Terwee, and Bridget Young. 2017. The COMET Handbook: Version 1.0.

Yuhao Zhang, Victor Zhong, Danqi Chen, Gabor Angeli, and Christopher D Manning. 2017. Position-aware attention and supervised data improve slot filling. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 35–45.

Appendices

A Hyperparameters and Run time

Using BioBERT in the Position based conditioning framework, we perform a grid search through multiple combinations of hyperparameters included in Table Table 5 below. The model is tuned on 20% of EBM-COMET dataset (as a dev set), we obtain the best Partial Match (PM) and Exact Match (EM) accuracies. Table Table 5 shows the range of values (including the lower and upper bound) for which the model is tuned to obtain optimal configurations. Using a shared TITAN RTX 24GB GPU, the baseline model runs for approximately 40 minutes per epoch.

Parameter	Tuned-range	Optimal
Train Batch size	[8,16,32]	16,32
Eval Batch size	[8,16,32]	8
Query type embedding size	[50,100,150]	50
Position embedding size	[100,200,300]	300
Attention layer size	[100,200,300]	200
Optimizer	[Adam, SGD]	Adam
Learning rate	[5e-5, 1e-4, 5e-3, 1e-3]	5e-5

Table 5: Parameter settings for the Position-based conditioning model

B Datasets

B.1 EBM-NLP

EBM-NLP corpus (Nye et al., 2018) is a crowd sourced dataset in which ca.5,000 clinical trial abstracts were annotated with elements in the health literature searching PICO framework (Huang et al., 2006). PICO stands for Participants, Interventions, Comparators and Outcomes. The dataset has supported clinicalNLP research tasks (Beltagy et al., 2019; Brockmeier et al., 2019). The corpus has two versions, (1) the "**starting spans**" in which text spans are annotated with the literal "PIO" labels (I and C merged into I) and (2) the "**hierarchical labels**" in which the annotated outcome "PIO" spans were annotated with more specific labels aligned to the concepts codified by the Medical Subject Headings (MeSH) [1], for instance the Outcomes (O) spans are annotated with more granular (specific) labels which include Physical, Pain, Mental, Mortality and Adverse effects. For the clinical recognition task we attempt, we use the hierarchical version of the dataset. The dataset has however

[1] https://www.nlm.nih.gov/mesh

been discovered to have flawed outcome annotations (Abaho et al., 2019) such as (1) statistical metrics and measurement tools annotated as part of clinical outcomes e.g."*mean arterial blood pressure*" instead of "*arterial blood-pressure*","*Quality of life Questionnaire*" instead of "*Quality of life*" and (2) Multiple outcomes annotated as a single outcome "*Systolic and Diastolic blood- pressure*" instead of "*Systolic blood-pressure*" and "*Diastolic blood-pressure*".

B.2 EBM-COMET

A biomedical corpus containing 300 PubMed "Randomised controlled Trial" abstracts manually annotated with outcome classifications drawn from the taxonomy proposed by (Dodd et al., 2018). The abstracts were annotated by two experts with extensive experience in annotating outcomes in systematic reviews of clinical trials (Abaho et al., 2021b). Dodd et al. (2018)'s taxonomy hierarchically categorised 38 outcome domains into 5 outcome core areas and applied this classification system to 299 published core outcome sets (COS) in the Core Outcomes Measures in Effectiveness (COMET) database.

C Layer probing

Initially, the hidden state we used (Equation (2)) extracted from the last layer for each of the Biomedical PLMs for all experiments. We however explore an option of extracting a weighted average of representation across all layers (Equation (12)) as a hidden state and study the performance of the models once this hidden state is introduced in the Position based conditioning framework to obtain position-aware representations.

$$\boldsymbol{h_i^l} = \text{PLM}_\theta(x_i) \tag{11}$$

$$\boldsymbol{h_i} = \text{MeanPool}(\boldsymbol{h_i^1}, .., \boldsymbol{h_i^l}, .., \boldsymbol{h_i^{l_N}}) \tag{12}$$

where h_i^l is a hidden state extracted from the l^{th} layer for word x at position i.

We only repeat training experiments using the Contextual PBC setup (subsection 3.6) however this time round using a mean pooled embedding across all layers as the hidden state. We notice that, aggregating a tokens representation by mean pooling across all layers of the transformer-based models does improve the performance in the outcome recalling experiments for both datasets.

Dataset	EBM-COMET			
Method	Contextual PBC (last layer)		Contextual PBC (Mean pool)	
Metric	EM	PM	EM	PM
BERT	43.32	55.94	45.80	57.19
BioBERT	53.34	59.65	53.58	61.22
SciBERT	63.00	70.95	63.15	72.67
Biomed_Roberta	44.32	62.86	45.00	63.17
UmlsBERT	31.88	36.46	33.10	39.21
Mean score	47.37	57.17	48.13	58.70

Table 6: Table reports EM and PM accuracies of the various biomedical Pre-trained Language Models for the outcome recalling experiments using the EBM-COMET and Contextual PBC. Mean score in a particular column is the average across all results in that column.

Dataset	EBM-NLP			
Method	Contextual PBC (last layer)		Contextual PBC (Mean pool)	
Metric	EM	PM	EM	PM
BERT	47.31	51.06	47.45	53.41
BioBERT	52.15	54.50	54.80	55.15
SciBERT	59,44	63.91	60.08	66.93
Biomed_Roberta	49.17	55.00	49.19	56.33
UmlsBERT	39.16	40.15	41.12	42.41
Mean score	49.45	52.92	50.53	54.85

Table 7: Table reports EM and PM accuracies of the various biomedical Pre-trained Language Models for the outcome recalling experiments using the EBM-NLP and Contextual PBC. Mean score in a particular column is the average across all results in that column.