Improving Contextual Representation with Gloss Regularized Pre-training

Anonymous ACL submission

Abstract

Though achieving impressive results on many NLP tasks, the BERT-like masked language models (MLM) encounter the discrepancy between pre-training and inference. In light of this gap, we investigate the contextual representation of pre-training and inference from the perspective of word probability distribution. We discover that BERT risks neglecting the contextual word similarity in pre-training. To tackle this issue, we propose an auxiliary gloss regularizer module to BERT pre-training (GR-BERT), to enhance word semantic similarity. By predicting masked words and aligning contextual embeddings to corresponding glosses simultaneously, the word similarity can be explicitly modeled. We design two architectures for GR-BERT and evaluate our model in downstream tasks. Experimental results show that the gloss regularizer benefits BERT in word-level and sentence-level semantic representation. The GR-BERT achieves new state-of-the-art in lexical substitution task and greatly promotes BERT sentence representation in both unsupervised and supervised STS tasks.

1 Introduction

Pre-trained language models like BERT (Devlin et al., 2019) and its variants (Liu et al., 2019b; Lan et al., 2019; Zhang et al., 2019; Joshi et al., 2020) have achieved remarkable success in a wide range of natural language processing (NLP) benchmarks. By pre-training on large scale unlabeled corpora, BERT-like models learn contextual representations with both syntactic and semantic properties. Researches show the contextual representations generated by BERT capture various linguistic knowledge, including part-of-speech, named entities, semantic roles (Tenney et al., 2019; Liu et al., 2019a; Ettinger, 2020), word senses (Wiedemann et al., 2019), etc. Furthermore, with the fine-tuning procedure, the contextual representations show excellent transferability in downstream language understanding tasks, and lead to state-of-the-art (SOTA) performance.

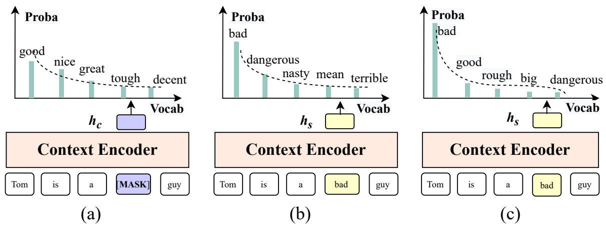


Figure 1: Conditional token probability distribution of tokens given masked context (a) and full context (b) and (c). The ideal token distribution given full context is illustrated in (b), while (c) shows the full contextual token distribution generated by actual BERT.

The masked language model (MLM) plays a significant role in the pre-training stage of many BERT-like models (Liu et al., 2019b). In an MLM, a token w is sampled from a text sequence $\mathbf{s}$, and replaced with a `[MASK]` token. Let $\mathbf{c}$ be the rest of tokens in $\mathbf{s}$ except for w. We name $\mathbf{c}$ as the *masked context* or *surrounding context*, and $\mathbf{s}$ as the *full context*. During pre-training, BERT encodes the masked context $\mathbf{c}$ into a contextual embedding vector $\boldsymbol{h}_c$, and use it to generate a contextual token probability distribution $p(x|\mathbf{c})$, where $x \in V$ and V denotes the token vocabulary. The training objective is to predict the masked token w by maximizing likelihood function $\log p(w|\mathbf{c})$. In the fine-tuning or inference stage, BERT takes the full context $\mathbf{s}$ without masks as input, and encodes every token into its contextual representationå for downstream tasks. We denote the contextual representation corresponds to token w as $\boldsymbol{h}_s$.

We analyze the corresponding contextual token probability distribution $p(x|\mathbf{c})$ and $p(x|\mathbf{s})$ generated from $\boldsymbol{h}_c$ and $\boldsymbol{h}_s$, as a proxy to study the representations (Li et al., 2020). Figure 1(a) shows an example when masked context $\mathbf{c}$ =“*Tom is a* `[MASK]` *guy*”, the predicted tokens with high probabilities $p(x|\mathbf{c})$ includes *good*, *nice*, *great*, *tough*, which are all reasonable answers to the Cloze task. Ideally, we want the context encoder to behave the same way when full context **s** is given, as in Figure 1(b), the model should only propose contextual synonyms of *bad* such as *dangerous*, *nasty* and *mean* with $p(x|\mathbf{s})$. However, the actual BERT generates $\hat{p}(x|\mathbf{s})$ as shown in Figure 1(c), which contains inappropriate token proposals such as *good*, *rough* and *big*.

The discrepancy between Figure 1(b) and 1(c) is because only the masked token distribution $p(x|\mathbf{c})$ is explicitly modeled in BERT with the MLM, while the full contextual token distribution $p(x|\mathbf{s})$ works in an agnostic way through model generalization. This leads to a gap between $p(x|\mathbf{c})$ in pre-training and $p(x|\mathbf{s})$ in fine-tuning and inference. It is shown in unsupervised scenarios, BERT generates contextual embeddings that even underperforms static embeddings for sentence representation (Reimers and Gurevych, 2019). Although in BERT pre-training, random token replacement strategy is used to mitigate the mismatch that `[MASK]` token is never seen during fine-tuning, to the best of the authors' knowledge, there is no analysis on the gap of representation between masked context $\boldsymbol{h}_c$ and full context $\boldsymbol{h}_s$ in different phases when using BERT.

To address this issue, we perform an investigation on the inner structure of $p(x|\mathbf{s})$. Through theoretical derivation, we discover $p(x|\mathbf{s})$ can be decomposed into the combination of masked contextual token distribution $p(x|\mathbf{c})$ and a point-wise mutual information (PMI) term that describes contextual token similarity. Further analysis shows both the MLM and token replacement in BERT pre-training have potential shortcomings in modeling the contextual token similarity. Inspired by the decomposition of $p(x|\mathbf{s})$, we propose to add an auxiliary gloss regularizer (GR) module to the MLM task, where mask prediction and gloss matching are trained simultaneously in the BERT pre-training. We also design two model architectures to integrate the gloss regularizer into the original MLM task.

We examine our proposed model in downstream tasks including unsupervised lexical substitution (LS) (McCarthy and Navigli, 2007; Kremer et al., 2014), semantic textual similarity (STS) and supervised STS Benchmark (Cer et al., 2017). By invoking gloss regularized pre-training, our model improves lexical substitution task from 14.5 to 15.2 points in the LS14 dataset, leading to new SOTA performance. In unsupervised STS tasks, gloss regularizer improves the performance from 56.57 to 67.47 in terms of average Spearman correlation by a large margin. Such performance gain is also observed in supervised STS task. Empirical experiments prove our model effectively generates better contextual token distribution and representations, which contributes to word-level and sentence-level language understanding tasks.

2 Related Works

Masked Language Models. Liu et al. (2019b) extend BERT into RoBERTa achieving substantial improvements. They claim the MLM task as the key contributor to contextual representation modeling, compared with next sentence prediction task. Many BERT variants focus on better masking strategies (Cui et al., 2019; Zhang et al., 2019; Joshi et al., 2020) to enhance the robustness and transferability of contextual representative learning. However, MLM suffers from the discrepancy between pre-training and fine-tuning since the `[MASK]` tokens are only introduced during pre-training. To tackle this issue, permutation language model from XLNet (Yang et al., 2019) and token replacement detection from ELECTRA (Clark et al., 2020) are proposed as alternative approaches to the MLM. Instead of avoiding MLM, we analyze how the mask modeling affects the full contextual representation in a probability perspective, and introduce gloss regularizer to mitigate the gap brought by MLM.

Contextual Representation Analysis. One way to analyze the contextual representation learned by pre-trained language model is through the probing tasks (Liu et al., 2019a; Miaschi and Dell'Orletta, 2020; Vulić et al., 2020), which are regarded as an empirical proofs that pre-trained MLMs like BERT succeed in capturing linguistic knowledge. Many other researches focus on studying the geometry of contextual representations. Ethayarajh (2019) discovers anisotropy among the contextual embeddings of words when studying contextuality of BERT. Li et al. (2020) propose a method using normalizing flow to transform the contextual embedding distribution of BERT into an isotropic distribution, and achieve performance gains in sentence-level tasks.

Utilizing Word Senses. Because the BERT conveys contextualized semantic knowledge of polysemous, many researches use BERT as a backbone to build word sense disambiguation (WSD) models (Huang et al., 2019; Blevins and Zettlemoyer, 2020; Bevilacqua and Navigli, 2020). In these models, BERT is used as word senses and contexts encoders to perform the downstream matching task. One work that directly incorporates word sense knowledge into pre-training is SenseBERT (Levine et al., 2020) that introduces a weakly-supervised supersense prediction task, which leads to improvement on performance of WSD and word-in-context task. In SenseBERT, word prediction is enhanced with supersense category labels that act like an external knowledge source. However, the gloss regularizer in our model provides fine-grained semantic information, which aimed to align word representation space with the semantic space, and leads to better contextual representations.

3 Contextual Token Probability

3.1 Masked Language Model

Without loss of generality, the token probability distribution given full context $p(x|\mathbf{s})$ can be decomposed into two parts,

$$\log p(x|\mathbf{s}) = \log p(x|\mathbf{c}) + \mathrm{PMI}(x; w|\mathbf{c}), \qquad (1)$$

where $\mathrm{PMI}(x; w|\mathbf{c})$ is the pointwise mutual information between x and w given $\mathbf{c}$. PMI describes how frequently two tokens co-occur than their independent occurrences, which is used as a measurement of the semantic similarity between tokens (Ethayarajh, 2019; Li et al., 2020). In Eqn. (1), $\log p(x|\mathbf{c})$ only depends on masked context, which directly corresponds to the MLM training objective. However, the PMI term is not explicitly modeled.

In BERT, $p(x|\mathbf{c})$ is generated from the encoded mask context $\boldsymbol{h}_c$ with a softmax operation as

$$p(x|\mathbf{c}) = \mathrm{softmax}(\boldsymbol{h}_c^\top \boldsymbol{v}_x), \qquad (2)$$

where $\boldsymbol{v}_x$ stands for the embedding vector of token x in vocabulary V. During fine-tuning or inference stage, full context $\mathbf{s}$ without masks is encoded into $\boldsymbol{h}_s$ as the contextual representation of token w. We can use the $\boldsymbol{h}_s$ to estimate $p(x|\mathbf{s})$ in the same way as Eqn. (2), denoted by $\hat{p}(x|\mathbf{s})$,

$$\hat{p}(x|\mathbf{s}) = \hat{p}(x|\mathbf{c}, w) = \mathrm{softmax}(\boldsymbol{h}_s^\top \boldsymbol{v}_x). \qquad (3)$$

Under such approximation setup, $\mathrm{PMI}(x; w|\mathbf{c})$ can be transformed into

$$\begin{aligned}\mathrm{PMI}(x; w|\mathbf{c}) &\approx \log \frac{\hat{p}(x|\mathbf{w}, c)}{p(x|\mathbf{c})} \\ &= (\boldsymbol{h}_s - \boldsymbol{h}_c)^\top \boldsymbol{v}_x + \varphi(w, \mathbf{c}), \qquad (4)\end{aligned}$$

where $\varphi(w, \mathbf{c})$ is constant w.r.t x. In a deep neural network parameterized model like BERT, $\boldsymbol{h}_s$ is encoded in an agnostic way. Thus, it's difficulty to further derive the PMI in Eqn. (4).

For a simpler case, if we consider a one-layer continuous bag-of-words (CBOW) model (Mikolov et al., 2013) [1], we have $\boldsymbol{h}_s - \boldsymbol{h}_c = \boldsymbol{h}_w$, where $\boldsymbol{h}_w$ is a context vector only related to the center token w. Now PMI is formulated as

$$\mathrm{PMI}_{\mathrm{CBOW}}(x; w|\mathbf{c}) = \log p(x|w) + \psi(w, \mathbf{c}),$$

where $\psi(w, \mathbf{c})$ is another constant w.r.t x. In this case, the PMI only contains similarity information between x and w, while the context information is completely ignored.

Although $\boldsymbol{h}_s - \boldsymbol{h}_c = \boldsymbol{h}_w$ is not satisfied in a deep model like BERT, the input sequences for $\boldsymbol{h}_s$ and $\boldsymbol{h}_c$ share the most identical tokens $\mathbf{c}$, and their only difference is whether to mask w. Therefore, there is a potential risk that $\mathrm{PMI}(x; w|\mathbf{c})$ in MLM loses information related to the condition $\mathbf{c}$, and degrades to the marginal $\mathrm{PMI}(x; w)$, especially when the MLM lacks modeling $p(x|\mathbf{s})$ in its training objective.

3.2 Replaced Language Model

In the BERT training process, a portion of tokens are replaced with random real tokens other than `[MASK]`, and the model is trained to predict the original tokens. We name this task as the replaced language model (RLM). Different from MLM, an RLM takes full context without masked tokens as input, and directly generates token distribution $p(x|\mathbf{s})$, which seems to be a better way for full contextual representation modeling.

We take a closer look at the RLM training process. Let $p(x|\mathbf{s}) = p(x|w, \mathbf{c})$ be the probability that token w is replaced with token x in context $\mathbf{c}$. According to the Bayes' theorem, we have

$$p(x|w, \mathbf{c}) = \frac{p(x|\mathbf{c})p(w|x, \mathbf{c})}{\sum_{x' \in V} p(x'|\mathbf{c})p(w|x', \mathbf{c})}. \qquad (5)$$

In a well-trained model, $p(w|x, \mathbf{c})$ should be the replacing probability during training. Since the process of replacing words by random noise is irrelevant to the context, $p(w|x, \mathbf{c}) = p(w|x)$. Let α be the probability when a token remains unchanged,

[1] The CBOW model can be considered as a kind of masked language model.

and $1-\alpha$ be the replacing probability. Therefore,

$$p(x|\mathbf{s}) = \frac{(1-\alpha)p(x|\mathbf{c})}{\alpha|V|p(w|\mathbf{c}) + (1-\alpha)\sum_{x'\neq w} p(x'|\mathbf{c})}, \tag{6}$$

where $|V|$ denotes the vocabulary size.

Eqn. (6) shows in RLM $p(x|\mathbf{s})$ is proportional to $p(x|\mathbf{c})$ and $\mathrm{PMI}(x;w|\mathbf{c})$ is constant when $x \neq w$, which means the distribution of x ($x \neq w$) only relies on surrounding context $\mathbf{c}$, but pays no attention to the center token w. This infers the RLM actually models the token distribution conditioning on almost only the surrounding context, even if it takes full context as input. Since the PMI term is completely ignored, RLM performs even worse the MLM in full contextual representation.

4 Gloss Regularizer

4.1 Invoking Gloss Matching

As shown in Eqn. (1), $p(x|\mathbf{s})$ consists of $p(x|\mathbf{c})$ and $\mathrm{PMI}(x;w|\mathbf{c})$. Both MLM and RLM succeed in modeling $p(x|\mathbf{c})$. However, the analysis in Section 3 shows RLM completely ignores $\mathrm{PMI}(x;w|\mathbf{c})$, and MLM may suffer from potential risks that the contextual information in $\mathrm{PMI}(x;w|\mathbf{c})$ would be lost, in either way the model generates poor estimation of $p(x|\mathbf{s})$.

$\mathrm{PMI}(x;w|\mathbf{c})$ describes co-occurrence probability of x and w normalized by their marginal probabilities under context $\mathbf{c}$ as condition. Ideally, it should be learned by training with labeled dataset $\{(\mathbf{s}_1, \mathbf{s}_2)\}$, where $\mathbf{s}_1 = \{x_1, \mathbf{c}\}$ and $\mathbf{s}_2 = \{x_2, \mathbf{c}\}$ are semantically similar text samples with shared context $\mathbf{c}$ and exchangeable token pair (x_1, x_2). However, such labeled data is expansive to build and not suitable for large-scale pre-training setup.

Intuitively, $\mathrm{PMI}(x;w|\mathbf{c})$ can be regarded as semantic similarity between tokens under context. Although the contexts of similar tokens are hard to obtain, we can use the glosses of tokens as an alternative. Since the semantic of a word can be defined by its gloss, contextual token similarity can be determined by detecting whether tokens are matching to similar glosses under context. Therefore, in order to better model the contextual token similarity defined by $\mathrm{PMI}(x;w|\mathbf{c})$, we introduce gloss matching an auxiliary task named the *gloss regularizer*.Two architectures to integrate gloss regularizer into MLM are detailed in Section 4.2 and 4.3.

4.2 Multi Task Model

A straight-forward method is to perform mask prediction and gloss matching as joint multitasks (denoted as MT). In this architecture, the masked context $\mathbf{c}$ and the full context $\mathbf{s}$ are encoded by a context encoder into the contextual vector $\boldsymbol{h}_c$ and $\boldsymbol{h}_s$. The loss function of the MLM task is

$$\mathcal{L}_{\text{MLM}} = -\boldsymbol{h}_c^\top \boldsymbol{v}_w + \log \sum_{w'\in V} \exp(\boldsymbol{h}_c^\top \boldsymbol{v}_{w'}). \tag{7}$$

For the gloss matching task, as illustrated in Figure 2(a), let $\mathbf{g}_t$ be the gloss text of token w under context $\mathbf{c}$. Another gloss encoder is used to encode $\mathbf{g}_t$ into a gloss vector $\boldsymbol{e}_t$. Gloss matching is performed by calculating the similarity between the contextual token representation $\boldsymbol{h}_s$ and the gloss vector $\boldsymbol{e}_t$. The gloss regularizing loss is

$$\mathcal{L}_{\text{GR}} = -\mathrm{sim}(\boldsymbol{h}_s, \boldsymbol{e}_t) + \log \sum_{t'\in T} \exp \mathrm{sim}(\boldsymbol{h}_s, \boldsymbol{e}_{t'}), \tag{8}$$

where $\mathrm{sim}(\cdot)$ is a similarity measurement function, and T is a set of negative glosses. The final loss function is the combination of the two losses,

$$\mathcal{L}_{\text{MT}} = \mathcal{L}_{\text{MLM}} + \lambda \mathcal{L}_{\text{GR}}, \tag{9}$$

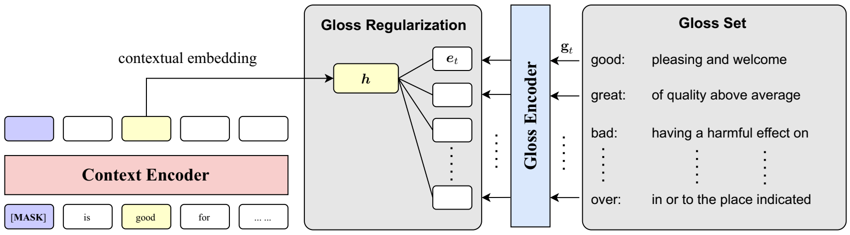
where λ denotes the regularizing weight.

This setting resembles the bi-encoders model (BEM) for WSD proposed by (Blevins and Zettlemoyer, 2020). However, in our model, the context encoder is trained on mask prediction task simultaneously with the gloss matching task, while the BEM takes gloss matching as a fine-tuning task. We train the two tasks together for better contextual and semantic representation modeling. As a result, the model learns token distribution not only conditioning on the masked context, but also influenced by semantic similarity with center token, which gives a better estimation of $p(x|\mathbf{s})$.

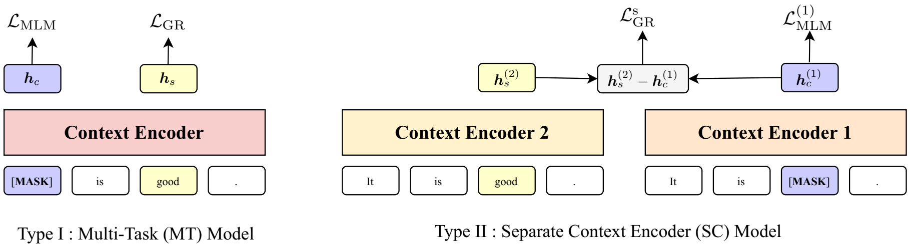
4.3 Separate Context Encoder Model

Another method is directly inspired by the decomposition from Eqn. (1). Different from the multitask model, we use two context encoders instead of one (denoted as SC). The first context encoder, denoted by $\mathtt{enc}_1$, encodes the masked context as $\boldsymbol{h}_c^{(1)} = \mathtt{enc}_1(\mathbf{c})$, and learns purely from MLM task with loss $\mathcal{L}_{\text{MLM}}^{(1)}$ derived similar as Eqn. (7).

The full context $\mathbf{s}$ is encoded into $\boldsymbol{h}_s^{(2)} = \mathtt{enc}_2(\mathbf{s})$ by the second encoder. Eqn. (4) shows



(a) The framework of the gloss regularized BERT



(b) Two types of the context encoder structures

Figure 2: (a) shows the gloss regularizer aligns contextual representation space with the gloss space. (b) Two GR architectures: the MT trains MLM and GR as multitask, while the SC utilizes two independent context encoders (the loss $\mathcal{L}^{(2)}_{\text{MLM}}$of SC is not shown).

$\text{PMI}(x; w|\mathbf{c})$ is entailed in the linear difference between the encoding of full and masked context. Therefore, we use $(\boldsymbol{h}_s^{(2)} - \boldsymbol{h}_c^{(1)})$ for gloss matching, where the loss function is formulated as

$$\mathcal{L}^{\text{s}}_{\text{GR}} = -\text{sim}\big[\boldsymbol{e}_t, \boldsymbol{h}_s^{(2)} - \boldsymbol{h}_c^{(1)}\big] + \log \sum_{t' \in T} \exp \text{sim}\big[\boldsymbol{e}_{t'}, \boldsymbol{h}_s^{(2)} - \boldsymbol{h}_c^{(1)}\big]. \quad (10)$$

In order to make the gloss matching learned by $\texttt{enc}_2$ aligned with the word embedding space, another MLM task is added to the training of $\texttt{enc}_2$, with loss $\mathcal{L}^{(2)}_{\text{MLM}}$. Thus, the complete loss function of the SC model is

$$\mathcal{L}_{\text{SC}} = \mathcal{L}^{(1)}_{\text{MLM}} + \mathcal{L}^{(2)}_{\text{MLM}} + \lambda \mathcal{L}^{\text{s}}_{\text{GR}}. \quad (11)$$

Although one gloss encoder and two contextual encoders are involved during training, only $\texttt{enc}_2$ is used at the inference stage. The contextual token distribution is given by $p(x|\mathbf{s}) = \text{softmax}(\boldsymbol{v}_w^\top \boldsymbol{h}_s^{(2)})$. By using two separate contextual encoders, the MLM task and gloss matching tasks can be trained individually, which leads to better performance for each task. Besides, the combination of the two tasks corresponds to the theoretical derivation of $p(x|\mathbf{s})$, and integrates the gloss regularizer in a more natural and explainable way.

4.4 Gloss Regularized Pre-training

Since we trained the contextual encoder and gloss encoder simultaneously, when evaluating the gloss matching loss, it is infeasible to encode the whole gloss set to calculate the full softmax. We thus use the in-batch negative sampling strategy from (Chen et al., 2017). Besides, we also use the other glosses of the target word as hard negatives for effective training.

We employ the gloss dataset from the online Oxford dictionary released by Chang et al. (2018); Chang and Chen (2019), formated in triplets of word, sentence and defination. The data consists of 677,191 pieces in total, including 31,889 words and 78,105 glosses. We utilize the BERT and RoBERTa model to initialize the context encoder and gloss encoder in our model. The pre-training settings and hyper-parameters are detailed in Appendix A.

5 Experiments

5.1 Downstream Tasks

In this section, we evaluate our model on three language understanding tasks. First, we choose the lexical substitution task to observe the word-level semantic performance. Then we conduct experiments on two sentence representation tasks: the STS task in unsupervised setting and the supervised STS benchmark (STS-B) task.

5.2 Lexical Substitution

Task and Dataset. Lexical substitution aims to replace the target word in a given context sentence

method	backbone	post process	SemEval 2007 (LS07)			CoInCo (LS14)		
			best/best-m	oot/oot-m	P@1/P@3	**best**/best-m	oot/oot-m	P@1/P@3
Roller and Erk (2016)	SGNS emb	-	-	-	19.7/14.8	-	-	18.2/13.8
Zhou et al. (2019)	$\text{BERT}_{\text{large}}$	-	12.1/20.2	40.8/56.9	13.1/-	9.1/19.7	33.5/56.9	14.3/-
		+valid	20.3/34.2	55.4/68.4	51.1/-	14.5/33.9	**45.9**/69.9	**56.3**/-
Arefyev et al. (2020)	$\text{RoBERTa}_{\text{large}}$	-	-	-	32.0/24.3	-	-	34.8/27.2
		+emb	-	-	44.1/31.7	-	-	46.5/36.3
	$\text{XLNet}_{\text{large}}$	+emb	-	-	49.5/34.9	-	-	51.4/39.1
baselines	$\text{BERT}_{\text{base}}$	-	13.2/22.3	40.8/57.1	33.1/23.7	10.1/21.9	33.0/56.5	38.4/28.7
	$\text{RoBERTa}_{\text{base}}$	-	16.7/27.8	45.2/62.9	40.8/28.5	11.0/23.6	34.9/59.3	42.2/31.4
our work	MT GR-$\text{BERT}_{\text{base}}$	-	17.7/30.8	49.8/67.8	42.5/31.1	12.2/ 26.5	39.2/64.5	46.4/35.3
	SC GR-$\text{BERT}_{\text{base}}$	-	18.2/31.2	49.9/67.6	44.1/31.2	12.4/ 27.1	39.8/65.5	46.6/35.8
	MT GR-$\text{RoBERTa}_{\text{base}}$	-	19.7/32.9	53.0/72.8	47.9/34.2	12.9/28.3	40.6/66.4	48.6/37.2
	SC GR-$\text{RoBERTa}_{\text{base}}$	-	19.4/33.2	52.8/71.5	47.4/33.4	13.1/28.8	40.9/66.6	48.8/37.8
		+emb	22.4/38.2	56.4/76.0	53.7/37.8	14.5/32.8	43.8/69.9	53.5/ 41.4
		+valid	22.6/38.4	56.0/73.9	54.8/39.0	15.1/33.7	44.1/69.6	56.0/42.7
		+both	**23.1/39.7**	**57.6/76.3**	**55.0/40.3**	**15.2/34.4**	45.3/**71.3**	55.9/**43.5**

Table 1: Comparison with previous SOTA on lexical substitution task. Results of the first three works are from the mentioned papers and the results in the baseline are from our experiments with the same word process.

by a substitute word that not only is semantically consistent with the original word but also preserves the sentence's meaning. There are two benchmark datasets for this task: the SemEval 2007 dataset (LS07) (McCarthy and Navigli, 2007) with 201 target words, and the CoInCo dataset (LS14) (Kremer et al., 2014) with 4,255 target words, both of which are unsupervised. The task LS07 releases the official evaluation metrics *best/best-mode* and *oot/oot-mode*[2], which evaluate the quality of the best prediction and the best 10 predictions, separately. We also report the metrics precision@1 (P@1) and P@3. Because the metric *best* considers the word frequencies in annotated labels, we take it as the main metric in this task.

Candidate Generation. We use the context encoder pre-trained with GR to generate lexical substitutions. Given a target word w and its context $\mathbf{s}$, we directly employ the full contextual token distribution $p(x|\mathbf{s})$ to perform the word prediction, then sort the candidates by their probabilities. We then lemmatize the word candidates as detailed in Appendix B.

Post-Process. Previous works proposed several effective approaches to improve LS performance. Arefyev et al. (2020) used the input word embedding to inject more target word information (noted +*emb*). Zhou et al. (2019) utilized a pre-trained model to re-score candidates (noted +*valid*). We denote these approaches as *post-process* and adopt them in our experiments. As Arefyev et al. (2020) reported, the result in (Zhou et al., 2019) is hardly reproduced and their code is not available, we then implement the validation process by ourselves.

Result and Analysis. Table 1 shows the comparison of our models with the previous SOTAs in LS07 and LS14 benchmarks. We first compare the model outputs without post-process. Our GR models surpass their MLM baselines by large margins in all metrics: the *best* value increases more than 3 points, the *oot* increases about 8 points in LS07. In separate context encoder structure, the *best* value of BERT increases from 10.1 to 12.4 in LS14, and the metric increases from 11.0 to 13.1 for RoBERTa. Comparing the P@1 with (Arefyev et al., 2020), the SC GR-RoBERTa base model 48.8 even exceeds the large RoBERTa model with *emb* 46.5.

Results indicate that GR model generates more semantically similar words and preserve the sentence original meaning even though no LS-like training data is used. This is because the gloss regularization plays the key role in modeling contextual token distribution $p(x|\mathbf{s})$ by taking both contextual and semantic information into consideration. Given a sentence context, if two words are semantically replaceable, their gloss text descriptions are naturally similar. As the word contextual embedding is aligned with its gloss, the words in semantically similar contexts are gathered closer indirectly, which benefits the LS task.

We further apply post-process on the SC GR-RoBERTa model. Consistent with previous works (Arefyev et al., 2020; Zhou et al., 2019), both processes improve the performance in testset LS14: $+emb$ increases the *best* value from 13.1 to 14.5, and it is to 15.1 using $+valid$. By applying

[2] http://www.dianamccarthy.co.uk/task10index.html

Model	STS12	STS13	STS14	STS15	STS16	STS-B	SICK-R	Avg.
GloVe embs	55.14	70.66	59.73	68.25	63.66	58.02	53.76	61.32
BERT-flow	58.40	67.10	60.85	75.16	71.22	68.66	64.47	66.55
BERT-whitening(NLI)	57.83	66.90	60.90	75.08	71.31	68.24	63.73	66.28
SimCSE-BERT	68.40	**82.41**	**74.38**	80.91	78.56	76.85	**72.23**	76.25
SimCSE-RoBERTa	**70.16**	81.77	73.24	**81.36**	**80.65**	**80.22**	68.56	**76.57**
BERT(first-last avg)	39.70	59.38	49.67	66.03	66.19	53.87	62.06	56.70
MT GR-BERT(first-last avg.)	53.20	**69.68**	58.81	**73.25**	**72.16**	**66.65**	**66.47**	**65.75**
SC GR-BERT(first-last avg.)	**53.69**	68.66	**58.83**	71.90	71.64	66.18	66.46	65.34
RoBERTa(first-last avg.)	40.88	58.74	49.07	65.63	61.48	58.55	61.63	56.57
MT GR-RoBERTa(first-last avg.)	**53.73**	**72.57**	**61.04**	**75.23**	**72.86**	69.44	**67.39**	**67.47**
SC GR-RoBERTa(first-last avg.)	53.69	70.00	59.24	72.38	72.47	**70.12**	67.02	66.42

Table 2: Sentence embedding performance on unsupervised STS tasks. Results in the first row are from Gao et al. 2021. Notation (first-last avg) means take the average of word embs from the input and output layer.

both post-processes, our SC GR-RoBERTa model achieves the new SOTA 15.2 in *best*. We also achieve SOTA in the metrics *best-m/oot-m* and P@3 in LS14 and all metrics in LS07. Appendix B demonstrates random selected examples of the LS task and the model outputs.

5.3 Unsupervised Sentence Representation Task

STS Task and Dataset. STS tasks deal with determining how similar two sentences are. We evaluate our model on 7 STS tasks: STS tasks 2012-2016 (Agirre et al., 2012, 2013, 2014, 2015, 2016), STS Benchmark (STS-B) (Cer et al., 2017) and SICK-Relatedness (SICK-R) (Marelli et al., 2014). Following the work of Gao et al. (2021) and their setting in STS tasks[3], we use *Spearman's correlation* with *"all" aggregation* as the evaluation metric, and use no additional regressor in experiments.

Baselines. Since our experiments are totally unsupervised: neither STS data nor NLI dataset[4] are used for training, we only perform comparison with previous works in unsupervised setting. SOTA works for these tasks are either trained by carefully designed sentence-level loss [e.g. SimCSE (Gao et al., 2021), BERT-flow (Li and Roth, 2002)] or tuned on sentence dataset NLI [e.g. BERT-whitening (Su et al., 2021)]. Therefore, these models are able to generate effective sentence representation. In contrast, our model is not trained with any sentence tasks, and we simply use the average of contextual word embeddings to represent sentence. Thus, it is not very fair to directly compare with the mentioned sentence encoders. We then focus more on the comparison with the original MLM.

Result and Analysis. Table 2 shows the results on STS tasks. With gloss regularization in pretraining, the average Spearman's correlation increases from 56.70 to 65.75 in BERT model and from 56.57 to 67.47 for RoBERTa. Though still far below the SimCSE SOTA performance, our model approaches the BERT-whitening and BERT-flow without any deliberately designed sentence-level tasks or transforming word distribution on domain data. Reimers and Gurevych (2019) report the unsupervised BERT embedding is infeasible for STS and performs even worse than GloVe embedding. Li et al. (2020) blame it on the anisotropic distribution of BERT word embeddings. Our experiments show great gains of GR-BERT in sentence embedding, proving the advantage of gloss regularized contextual representation is also valid for sentences. A brief analysis on sentence representation with gloss regularizer is provided in Appendix C.

5.4 Supervised STS

STS-B Task and Dataset. We validate our model in supervised STS Benchmark (STS-B) (Cer et al., 2017). The data consists of 8,628 sentence pairs and is divided into trainset (5,749), devset (1,500) and testset (1,379).

Since supervised STS performance are largely influenced by the training data, we only use the STS trainset in all experiments. Besides, we randomly reduce the data size to simulate the limit data scenarios and compare our model with MLM baselines. Following the sentence-BERT (Reimers

[3] https://github.com/princeton-nlp/SimCSE

[4] NLI dataset consists of SNLI and MNLI, both of which are proved to be effective domain data for STS tasks (Gao et al., 2021; Reimers and Gurevych, 2019).

Data ratio	Models	Spearman
100%	BERT	83.98 ± 0.16
	MT GR-BERT	$\mathbf{85.13} \pm 0.06$
	SC GR-BERT	85.00 ± 0.16
100%	RoBERTa	85.90 ± 0.57
	MT GR-RoBERTa	$\mathbf{86.87} \pm 0.21$
	SC GR-RoBERTa	86.25 ± 0.30
50%	BERT	81.60 ± 0.28
	MT GR-BERT	$\mathbf{83.47} \pm 0.15$
	SC GR-BERT	83.06 ± 0.19
20%	BERT	76.43 ± 0.37
	MT GR-BERT	$\mathbf{79.87} \pm 0.41$
	SC GR-BERT	79.18 ± 0.21

Table 3: Evaluation on STS-B test set. All experiments are fine-tune for 4 epochs with batch size 16. Results are the average of 4 random seeds.

model	LS14	STS Avg	STS-B
BERT	10.1	56.70	83.98
+MLM	10.9	62.22	84.62
MT GR-BERT	12.2	65.75	85.13
SC GR-BERT	12.4	65.34	85.00

Table 4: Ablation studies of different training loss in three tasks. +MLM means only use MLM loss in training. We use the metric *best* for LS14 task, the average Spearman's correlation for 7 STS tasks and STS-B.

and Gurevych, 2019)[5], we use Siamese BERT network with cosine similarity.

Result and Analysis. Tabel 3 shows the comparison on STS-B. In both BERT and RoBERTa backbones, GR models improve the baselines by around 0.9 points. In low-resource scenarios, the advantage of GR-BERT increases. When 50% data is available, the gain of MT GR-BERT is increased to 1.87 points, and the gain is up to 3.44 points for 20% data. Results show that in fine-tuning process, the GR model still preserves its advantage over MLM baselines in sentence semantic representation, indicating the contextual representation pre-trained with GR is transferable in further fine-tuning. The GR pre-training is able to enhance the semantic knowledge in model, especially in the low-resource data scenarios, which ease the hunger for task training data.

5.5 Ablation Analysis

We now investigate the influence of gloss training data and the model structures. Results are shown in Table 4. Gururangan et al. (2020) reports the domain data pre-training can improve model performance. To evaluate the influence of dictionary corpus, we pre-train BERT by MLM in the same dataset and find that high-quality data improves all three task performances. However, GR still contributes to the large part of the improvement, especially in the LS task. As for the two proposed structures, the SC-GR utilizes individual context encoders that impose less restriction on gloss learning, and achieves better performance in LS word-level task. On the contrary, the MT model provides a better sentence embedding and surpasses SC structure in STS tasks.

6 Conclusion

In this work, we propose the GR-BERT, a model with gloss regularization to enhance the word contextual information. We first analyze the gap between MLM pre-training and inference, and aim to model the PMI term that characterizes the word semantic similarity given context. Due to the lack of data that labels the word semantic similarities given contexts, we propose to indirectly learn the semantic information in pre-training by aligning contextual word embedding space to a human annotated gloss space. We design two model structures and validate them in three NLP semantic tasks. In the lexical substitution task, we increase the SOTA value from 14.5 to 15.2 in LS14 *best* metric and many other metrics in LS07 and LS14 are also improved. In the unsupervised STS task, our GR model show its capacity in sentence representation without any training in sentence task, and it improves the MLM performance from 56.57 to 67.47. In the supervised STS-B task, GR model exceed the MLM baseline by about 0.9 points, and the gains increases to 3.44 in the low resource scenarios.

Our work provides a new perspective to the MLM pre-training, and show the effectiveness of modeling word semantic similarity. However, one limitation of our work is the lack of large-scale word-gloss matching data. The training data in our work is far less than that in BERT pre-training. Our future works will focus on mining larger scale word-gloss training data and also validate GR model in more NLP tasks. We believe there is still a big room for GR model performance improvement and possible gains in more NLP tasks.

[5] https://www.sbert.net/examples/training/sts/README.html

References

Eneko Agirre, Carmen Banea, Claire Cardie, Daniel Cer, Mona Diab, Aitor Gonzalez-Agirre, Weiwei Guo, Iñigo Lopez-Gazpio, Montse Maritxalar, Rada Mihalcea, German Rigau, Larraitz Uria, and Janyce Wiebe. 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, Denver, Colorado. Association for Computational Linguistics.

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, Dublin, Ireland. Association for Computational Linguistics.

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. 2012. 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.

Nikolay Arefyev, Boris Sheludko, Alexander Podolskiy, and Alexander Panchenko. 2020. Always keep your target in mind: Studying semantics and improving performance of neural lexical substitution. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 1242–1255, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Michele Bevilacqua and Roberto Navigli. 2020. Breaking through the 80% glass ceiling: Raising the state of the art in word sense disambiguation by incorporating knowledge graph information. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2854–2864, Online. Association for Computational Linguistics.

Terra Blevins and Luke Zettlemoyer. 2020. Moving down the long tail of word sense disambiguation with gloss informed bi-encoders. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1006–1017, Online. Association for Computational Linguistics.

Daniel Cer, Mona Diab, Eneko Agirre, Iñigo Lopez-Gazpio, and Lucia Specia. 2017. 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.

Ting-Yun Chang and Yun-Nung Chen. 2019. What does this word mean? explaining contextualized embeddings with natural language definition. 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 6064–6070, Hong Kong, China. Association for Computational Linguistics.

Ting-Yun Chang, Ta-Chung Chi, Shang-Chi Tsai, and Yun-Nung Chen. 2018. xSense: Learning sense-separated sparse representations and textual definitions for explainable word sense networks. *arXiv preprint arXiv:1809.03348*.

Ting Chen, Yizhou Sun, Yue Shi, and Liangjie Hong. 2017. On sampling strategies for neural network-based collaborative filtering. In *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 767–776.

Kevin Clark, Minh-Thang Luong, Quoc V. Le, and Christopher D. Manning. 2020. ELECTRA: Pre-training text encoders as discriminators rather than generators. In *ICLR*.

Yiming Cui, Wanxiang Che, Ting Liu, Bing Qin, Ziqing Yang, Shijin Wang, and Guoping Hu. 2019. Pre-training with whole word masking for Chinese BERT. *arXiv preprint arXiv:1906.08101*.

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.

Kawin Ethayarajh. 2019. How contextual are contextualized word representations? Comparing the geometry of BERT, ELMo, and GPT-2 embeddings. 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 55–65, Hong Kong, China. Association for Computational Linguistics.

Allyson Ettinger. 2020. What BERT is not: Lessons from a new suite of psycholinguistic diagnostics for language models. *Transactions of the Association for Computational Linguistics*, 8:34–48.

Tianyu Gao, Xingcheng Yao, and Danqi Chen. 2021. SimCSE: Simple contrastive learning of sentence embeddings. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 6894–6910, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

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.

Luyao Huang, Chi Sun, Xipeng Qiu, and Xuanjing Huang. 2019. GlossBERT: BERT for word sense disambiguation with gloss knowledge. 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 3509–3514, Hong Kong, China. Association for Computational Linguistics.

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

Gerhard Kremer, Katrin Erk, Sebastian Padó, and Stefan Thater. 2014. What substitutes tell us - analysis of an "all-words" lexical substitution corpus. In *Proceedings of the 14th Conference of the European Chapter of the Association for Computational Linguistics*, pages 540–549, Gothenburg, Sweden. Association for Computational Linguistics.

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

Yoav Levine, Barak Lenz, Or Dagan, Ori Ram, Dan Padnos, Or Sharir, Shai Shalev-Shwartz, Amnon Shashua, and Yoav Shoham. 2020. SenseBERT: Driving some sense into BERT. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4656–4667, Online. Association for Computational Linguistics.

Bohan Li, Hao Zhou, Junxian He, Mingxuan Wang, Yiming Yang, and Lei Li. 2020. On the sentence embeddings from pre-trained language models. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 9119–9130, Online. Association for Computational Linguistics.

Xin Li and Dan Roth. 2002. Learning question classifiers. COLING '02, page 1–7, USA. Association for Computational Linguistics.

Nelson F. Liu, Matt Gardner, Yonatan Belinkov, Matthew E. Peters, and Noah A. Smith. 2019a. Linguistic knowledge and transferability of contextual representations. 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 1073–1094, Minneapolis, Minnesota. Association for Computational Linguistics.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019b. RoBERTa: A Robustly Optimized BERT Pretraining Approach.

Marco Marelli, Stefano Menini, Marco Baroni, Luisa Bentivogli, Raffaella Bernardi, and Roberto Zamparelli. 2014. A SICK cure for the evaluation of compositional distributional semantic models. In *Proceedings of the Ninth International Conference on Language Resources and Evaluation (LREC'14)*, pages 216–223, Reykjavik, Iceland. European Language Resources Association (ELRA).

Diana McCarthy and Roberto Navigli. 2007. SemEval-2007 task 10: English lexical substitution task. pages 48–53.

Alessio Miaschi and Felice Dell'Orletta. 2020. Contextual and non-contextual word embeddings: an in-depth linguistic investigation. In *Proceedings of the 5th Workshop on Representation Learning for NLP*, pages 110–119, Online. Association for Computational Linguistics.

Tomas Mikolov, Kai Chen, Greg Corrado, and Jeffrey Dean. 2013. Efficient estimation of word representations in vector space. *1st International Conference on Learning Representations, ICLR 2013 - Workshop Track Proceedings*, pages 1–12.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using siamese bert-networks. *arXiv preprint arXiv:1908.10084*.

Stephen Roller and Katrin Erk. 2016. PIC a different word: A simple model for lexical substitution in context. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1121–1126, San Diego, California. Association for Computational Linguistics.

Jianlin Su, Jiarun Cao, Weijie Liu, and Yangyiwen Ou. 2021. Whitening sentence representations for better semantics and faster retrieval. *arXiv preprint arXiv:2103.15316*.

Ian Tenney, Patrick Xia, Berlin Chen, Alex Wang, Adam Poliak, R Thomas McCoy, Najoung Kim, Benjamin Van Durme, Samuel R Bowman, Dipanjan Das, et al. 2019. What do you learn from context? probing for sentence structure in contextualized word representations. *arXiv preprint arXiv:1905.06316*.

Ivan Vulić, Edoardo M. Ponti, Robert Litschko, Goran Glavaš, and Anna Korhonen. 2020. Probing pretrained language models for lexical semantics. *EMNLP 2020 - 2020 Conference on Empirical Methods in Natural Language Processing, Proceedings of the Conference*, pages 7222–7240.

Gregor Wiedemann, Steffen Remus, Avi Chawla, and Chris Biemann. 2019. Does BERT make any sense? interpretable word sense disambiguation with contextualized embeddings. *arXiv preprint arXiv:1909.10430*.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Ruslan Salakhutdinov, and Quoc V. Le. 2019. XLNet: Generalized autoregressive pretraining for language understanding. In *Proceedings of the 33rd International Conference on Neural Information Processing Systems*, Red Hook, NY, USA. Curran Associates Inc.

Zhengyan Zhang, Xu Han, Zhiyuan Liu, Xin Jiang, Maosong Sun, and Qun Liu. 2019. ERNIE: Enhanced language representation with informative entities. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1441–1451, Florence, Italy. Association for Computational Linguistics.

Wangchunshu Zhou, Tao Ge, Ke Xu, Furu Wei, and Ming Zhou. 2019. BERT-based lexical substitution. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3368–3373, Florence, Italy. Association for Computational Linguistics.

A Pre-training Details

We employ the BERT-base uncased model and RoBERTa-base model to initialize the context and gloss encoders in our experiments. Both models are pre-trained on released Oxford dictionary data for around 10 epochs. We evaluate the model every epoch by the gloss matching accuracy on the randomly picked evaluation set. In the pre-training process, we set the GR loss weight as $\lambda = 2.0$. Following the SimCSE training hyper-parameters (Gao et al., 2021), we use cosine similarity between gloss embedding and contextual word embedding, and we set the temperature $\tau = 0.05$ in softmax. Take the MT GR model as an example, the softmax of gloss matching is $\text{softmax}(\text{cosine}(\boldsymbol{h}_s, \boldsymbol{e}_t)/\tau)$.

We conduct the pre-training on 8 Tesla V100 GPUs. The learning rate is set as 2×10^{-5}. The batch size for BERT is 48×8, and it is 36×8 for RoBERTa model.

B Lexical Substitution Details

As Arefyev et al. (2020) reported, the process on the format of word candidates influences the metrics. We thus (almost) follow their code[6] and fix the word process in all experiments. In our experiments, the word process includes lemmatization (*went->go*), filtering the candidates having the same lemmatization output with the original word and removing duplicate lemmatization of candidates. We also filter out the candidates according to the parts-of-speech (POS) information. For example, the word *good* can be used as *noun* or *adj*, but it would be unreasonable to serve as *verb*. We then check the possible POSs for each candidate and filter those words with unmatched POS with the target word.

In the post-process, the hyper-parameters in (+emb) and validation are tuned in LS07 data. Follow the implementation of Arefyev et al. (2020), we use cosine similarity and the temperature for similarity is set $1/15$ in all our experiments. For the validation process, we follow the idea of Zhou et al. (2019), but use BERT-base uncased model for validation. Following their work, we pick the first 50 candidates to re-rank (it has little influence when the number is above 20 in our experiments). The values in propose and validate scores are in different scales, as one is from logits and the other is from cosine similarity. We then adjust the weight of propose score to let its standard deviation be in the same level with the cosine similarity. We set the weight as 0.009 for RoBERTa and 0.004 for BERT.

Table 5 gives examples of LS task and compares our model outputs with the baseline.

C Sentence Similarity

We extend the contextual token similarity measurement into sentence similarity. As stated in (Li et al., 2020), the dot product similarity between sentence representations $\boldsymbol{h}_c^\top \boldsymbol{h}_{c'}$ is difficult to derived theoretically, since it is not explicitly involved in the BERT pre-training process. Therefore, inspired by token-level lexical substitution task using contextual probability distribution, we consider the probability distribution of a sentence $\mathbf{s}_1$ given another sentence $\mathbf{s}_2$, i.e. $p(\mathbf{s}_1|\mathbf{s}_2)$.

[6] https://github.com/Samsung/LexSubGen

target word	tell
sentence	He held Obi-Wan loosely , gently stroking his back He knew now that it did n't matter what Sampris said , or what Yoda **told** him .
labels	said to (4), inform (2)
RoBERTa	teach, say, give, call, have
SC GR-RoBERTa	teach, say, warn, instruct, promise
+ post-process	inform, teach, warn, say, instruct
target word	think
sentence	Shafer **thinks** we're going to cry , "he doesn't get it!" in reply to his piece" "it" being the amazing world of the Web and new media .
labels	believe (3), feel (1), suspect (1), reckon (1), assume (1)
RoBERTa	say, know, hop, believe, worry
SC GR-RoBERTa	believe, say, hop, expect, suspect
+ post-process	believe, say, hop, expect, know
target word	thus
sentence	The kind of control he exercises is **thus** likely to be limited to " passive " control such as inspection of produced goods and testing to insure that quality standards are being met .
labels	therefore (5), accordingly (1), consequently (1)
RoBERTa	typically, therefore, then, so, similarly
SC GR-RoBERTa	therefore, consequently, so, accordingly, hence
+ post-process	therefore, consequently, hence, thereby, so
target word	clean
sentence	Dog and horse owners should be encouraged to **clean** up after their animals .
labels	scrape (1), clear (2), tidy (2)
RoBERTa	wash, pick, wake, keep, clear
SC GR-RoBERTa	groom, walk, look, care, do
+ post-process	tidy, wash, groom, care, walk
target word	late
sentence	We were **late** doing this since I refused to use someone else 's " shopping cart " system that I did n't write and could n't trust .
labels	delayed (3), tardy (2), behind schedule (1), behind time (1), behind (1)
RoBERTa	also, early, just, still, already
SC GR-RoBERTa	early, slow, not, long, behindo
+ post-process	early, slow, prematurely, long, not
target word	new
sentence	The lecture itself went well , but a **new** problem arose .
labels	different (1), extra (1), additional (1), fresh (4)
RoBERTa	different, big, small, fresh, great
SC GR-RoBERTa	fresh, big, previous, further, different
+ post-process	fresh, renewed, different, previous, recent

Table 5: Examples from LS07 benchmark to show the task and model outputs. The number follows each label is the frequency count indicating the number of annotators that provided this substitute. For each model, we report the top 5 candidates in the first 50 predictions in lemmatized form.

Proposition 1. Let $w_1, \cdots, w_n$ be n tokens sampled from a sentence $\mathbf{s}$, and $\mathbf{c}_i$ be the rest of tokens in $\mathbf{s}$ except for w_i. Let $x_1, \cdots, x_n$ denote the tokens that can replace $w_1, \cdots, w_n$ in $\mathbf{s}$, respectively. The joint probability distribution of $x_1, \cdots, x_n$ given $\mathbf{s}$ is formulated as

$$\log p(x_1, \ldots, x_n|\mathbf{s}) = \sum_{i=1}^{n} P_i, \qquad (12)$$

where

$$P_i = \log p(x_i|\mathbf{c}_i, x_{<i}) + \mathrm{PMI}(x_i; w_i|\mathbf{c}_i, x_{<i}), \qquad (13)$$

and $x_{<i}$ denotes $x_1, \cdots, x_{i-1}$.

Proof We use the mathematical induction to proof the proposition.

When $n = 1$, $\log p(x_1|\mathbf{s}) = P_1$ is equivalent as Eqn. (1).

When $n > 1$, we make an assumption that Eqn. (12) holds true for $n = k-1$, i.e. $\log p(x_{<k}|\mathbf{s}) = \sum_{i=1}^{k-1} P_i$. Then,

$$\begin{aligned}
&\log p(x_{<k}, x_k|\mathbf{s}) \\
=&\log p(x_k|\mathbf{c}_k, x_{<k}) + \log \frac{p(x_k|w_k, \mathbf{c}_k, x_{<k})}{p(x_k|\mathbf{c}_k, x_{<k})} \cdots \\
&\qquad + \log \frac{p(x_k, x_{<k}|w_k, \mathbf{c}_k)}{p(x_k|w_k, \mathbf{c}_k, x_{<k})} \\
=&\log p(x_k|\mathbf{c}_k, x_{<k}) + \mathrm{PMI}(x_k; w_k|\mathbf{c}_k, x_{<k}) \cdots \\
&\qquad + \log p(x_{<k}|\mathbf{s}) \\
=&P_k + \sum_{i=1}^{k-1} P_i = \sum_{i=1}^{k} P_i, \qquad (14)
\end{aligned}$$

which means Eqn. (12) is also true for $n = k$. $\square$

Proposition 1 indicates one sentence can be transformed into another sentence through a series of token substitution operations, and the sentence transforming probability can be decomposed into the sum of a series of contextual token probabilities and contextual token similarities, i.e.

$$p(\mathbf{s}_1|\mathbf{s}_2) = \sum_{i=1}^{n} P_i, \qquad (15)$$

where P_i is defined in Eqn. (13), and $\mathbf{s}_1 = [x_1, \cdots, x_n], \mathbf{s}_2 = [w_1, \cdots, w_n]$. We ignore the case when $\mathbf{s}_1$ and $\mathbf{s}_2$ have different lengths, since a simple solution is to pad the shorter sentence to the length of the longer one.

Eqn. (15) and (13) show that the sentence-level tasks also benefits from our gloss regularizer, since the contextual token similarity modeled by gloss matching task also contributes to sentence representation.