Contextual Representation Learning beyond Masked Language Modeling

Zhiyi Fu[1*†], **Wangchunshu Zhou**[2*], **Jingjing Xu**[3†], **Hao Zhou**[2], **Lei Li**[3†]
[1]Peking University [2]ByteDance AI Lab [3]University of California, Santa Barbara
ypfzy@pku.edu.cn
{zhouwangchunshu.7, zhouhao.nlp}@bytedance.com
{jingjingxu, leili}@cs.ucsb.edu

Abstract

How do masked language models (MLMs) such as BERT learn contextual representations? In this work, we analyze the learning dynamics of MLMs. We find that MLMs adopt sampled embeddings as anchors to estimate and inject contextual semantics to representations, which limits the efficiency and effectiveness of MLMs. To address these issues, we propose TACO, a simple yet effective representation learning approach to directly model global semantics. TACO extracts and aligns contextual semantics hidden in contextualized representations to encourage models to attend global semantics when generating contextualized representations. Experiments on the GLUE benchmark show that TACO achieves up to 5x speedup and up to 1.2 points average improvement over existing MLMs. The code is available at https://github.com/FUZHIYI/TACO.

1 Introduction

In the age of deep learning, the basis of representation learning is to learn distributional semantics. The target of distributional semantics can be summed up in the so-called distributional hypothesis (Harris, 1954): *Linguistic items with similar distributions have similar meanings*. To model similar meanings, traditional representation approaches (Mikolov et al., 2013; Pennington et al., 2014) (e.g., Word2Vec) model distributional semantics by defining tokens using *context-independent* (CI) dense vectors, i.e., word embeddings, and directly aligning the representations of tokens in the same context. Nowadays, pre-trained language models (PTMs) (Devlin et al., 2019; Radford et al., 2018; Qiu et al., 2020) expand static embeddings into contextualized representations where each token has two kinds of representations: *context-independent* embedding, and *context-dependent* (CD) dense representation that stems from its embedding and contains context information. Although language modeling and representation learning have distinct targets, masked language modeling is still the prime choice to learn token representations with access to large scale of raw texts (Peters et al., 2018; Devlin et al., 2019; Raffel et al., 2020; Brown et al., 2020).

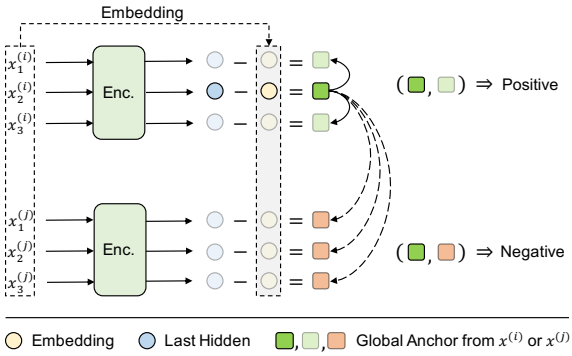

Figure 1: Illustration of the proposed token-alignment contrastive objective. It extracts and aligns the global semantics hidden in contextualized representations via the gap between contextualized representations and corresponding static embeddings.

It naturally raises a question: How do masked language models learn contextual representations? Following the widely-accepted understanding (Wang and Isola, 2020), MLM optimizes two properties, the alignment of contextualized representations with the static embeddings of masked tokens, and the uniformity of static embeddings in the representation space. In the alignment property, sampled embeddings of masked tokens play as an *anchor* to align contextualized representations. We find that although such local anchor is essential to model local dependencies, the lack of global anchors brings several limitations. First, experiments show that the learning of contextual representations is sensitive to embedding quality, which harms the efficiency of MLM at the early stage of

*Equal Contribution
†This work is done at ByteDance AI Lab.

training. Second, MLM typically masks multiple target words in a sentence, resulting in multiple embedding anchors in the same context. This pushes contextualized representations into different clusters and thus harms modeling global dependencies.

To address these challenges, we propose a novel **T**oken-**A**lignment **C**ontrastive **O**bjective (**TACO**) to directly build global anchors. By combing local anchors and global anchors together, TACO achieves better performance and faster convergence than MLM. Motivated by the widely-accepted belief that contextualized representation of a token should be the mapping of its static embedding on the contextual space given global information, we propose to directly align global information hidden in contextualized representations at all positions of a natural sentence to encourage models to attend same global semantics when generating contextualized representations. Concerning possible relationships between context-dependent and context-independent representations, we adopt the simplest probing method to extract global information via the gap between context-dependent and context-independent representations of a token for simplification, as shown in Figure 1. To be specific, we define tokens in the same context (text span) as positive pairs and tokens in different contexts as negative pairs, to encourage the global information among tokens within the same context to be more similar compared to that from different contexts.

We evaluate TACO on GLUE benchmark. Experiment results show that TACO outperforms MLM with average 1.2 point improvement and 5x speedup (in terms of sample efficiency) on BERT-small, and with average 0.9 point improvement and 2x speedup on BERT-base.

The contributions of this paper are as follows.

- We analyze the limitation of MLM and propose a simple yet efficient method TACO to directly model global semantics.
- Experiments show that TACO outperforms MLM with up to 1.2 point improvement and up to 5x speedup on GLUE benchmark.

2 Understanding Language Modeling

2.1 Objective Analysis

The key idea of MLM is to randomly replace a few tokens in a sentence with the special token `[MASK]` and ask a neural network to recover the original tokens. Formally, we define a corrupted sentence as $\boldsymbol{x}_1, \boldsymbol{x}_2, \cdots, \boldsymbol{x}_L$, and feed it into a Transformers encoder (Vaswani et al., 2017), the hidden states from the final layer are denoted as $\boldsymbol{h}_1, \boldsymbol{h}_2, \cdots, \boldsymbol{h}_L$. We denote the embeddings of the corresponding original tokens as $\boldsymbol{e}_1, \boldsymbol{e}_2, \cdots, \boldsymbol{e}_L$. The MLM objective can be formulated as:

$$\mathcal{L}_{\text{MLM}}(\boldsymbol{x}) = -\frac{1}{|\mathcal{M}|} \sum_{i \in \mathcal{M}} \log \frac{\exp(\boldsymbol{m}_i \cdot \boldsymbol{e}_i)}{\sum_{k=1}^{|\mathcal{V}|} \exp(\boldsymbol{m}_i \cdot \boldsymbol{e}_k)} \quad (1)$$

where $\mathcal{M}$ denotes the set of masked tokens and $|\mathcal{V}|$ is the size of vocabulary $\mathcal{V}$. $\boldsymbol{m}_i$ is hidden state of the last layer at the masked position, and can be regarded as a fusion of contextualized representations of surrounding tokens. Following the widely-accepted understanding (Wang and Isola, 2020), Eq.1 optimizes: (1) the alignment between contextualized representations of surrounding tokens and the context-independent embedding of the target token and (2) the uniformity of representations in the representation space.

In the alignment part, MLM relies on sampled contextual-independent embeddings of masked tokens as anchors to align contextualized representations in contexts, as shown in Figure 2. Local anchor is the key feature of MLM. Therefore, the learning of contextualized representations heavily relies on embedding quality. In addition, multiple local anchors in a sentence tend to pushing contextualized representations of surrounding tokens closer to different clusters, encouraging models to attend local dependencies where global semantics are neglected.

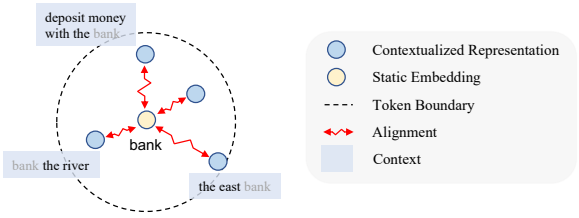

Figure 2: Illustration of the MLM objective. At the alignment part, it uses static embedding of masked tokens to align contextualized representations in the same context.

2.2 Empirical Analysis

To verify our understanding, we conduct comprehensive experiments to investigate: How does embedding anchor affect the learning dynamics of MLM? We re-train a BERT-small (Devlin et al., 2019) model with the MLM objective solely and analyze the changes in its semantic space during

pre-training. The training details are described in Appendix A.

Contextualized representation evaluation. In general, if contextualized representations are well learned, the contextualized representations in a same context will have higher similarity than that of in different contexts. Naturally, we use the gap between intra-sentence similarity and inter-sentence similarity to evaluate contextual information in contextualized representations. We call this gap as *contextual score*. The similarity can be evaluated via probing methods like L2 distance, cosine similarity, etc. We observe similar findings on different probing methods and only report cosine similarity here for simplification. Figure 3(b) shows how contextual score changes during training. Other statistical results are listed in Appendix A.

Embedding similarity evaluation. To observe how sampled embeddings affect contextualized representation learning, we evaluate the embedding similarity between co-occurrent tokens. Motivated by the target that co-occurrent tokens should have similar representations, we use the similarity score calculated by cosine similarity between co-occurrent words labeled by humans (sampled from the WordSim353 dataset (Agirre et al., 2009)) as the evaluation metric. Figure 3(a) shows how embedding similarity between co-occurrent tokens changes during training.

The learning of contextualized representations heavily relies on embeddings similarity. As we can see from Figure 3(a), the embedding similarity between co-occurrent tokens first decreases during the earliest stage of pre-training. It is because all embeddings are randomly initialized with the same distribution and the uniformity feature in MLM pushes tokens far away from each other, thus resulting in the decrease of embedding similarity. Meanwhile, the contextual score, i.e., the gap between intra-context similarity and inter-context similarity in Figure 3(b), does not increase at the earliest stage of training. It shows that random embeddings provide little help to learn contextual semantics. During 5K-10K iterations, only when embeddings become closer, contextualized representations in the same context begin to have similar features. At this stage, the randomly sampled embeddings from the same sentence, i.e., the same context, usually have similar representations and thus MLM can push contextualized tokens closer to each other.

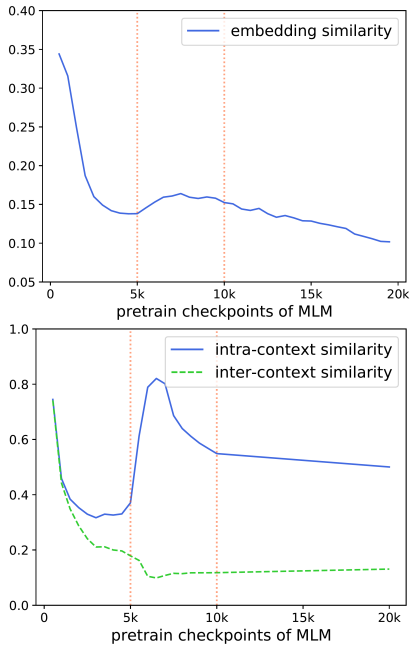

Figure 3: The learning dynamics of MLM. The top figure (a) illustrates the similarity between embeddings of frequently co-occurrent tokens (e.g., bank and money). The bottom figure (b) illustrates the similarity between contextualized representation of tokens from the same context and different contexts. These figures show an embedding bias problem where only the randomly selected target embeddings in MLM are similar, contextualized representations in the same context will be aligned with similar features.

We further verify the effects of embedding quality in Figure 4. To this end, we train two BERT models whose embedding matrices are frozen and initialized with the ones from different pre-training stage. We can see the model initialized with random embedding fails to teach contextualized representations to attend sentence meanings and representations from different contexts have almost the same similarity. However, the variant with well-trained but frozen embeddings learns to distinguish different contexts early at around 4k steps. These statistical observations verify that embedding anchors bring the efficiency and effectiveness problem.

Surprisingly, embedding anchors reduce global contextual information in contextualized representation at the later stage of training. Figure 3(a) shows that embedding similarity begins to drop after 8k steps. It shows that the model learns the specific meanings of co-occurrent tokens and begins to push them a little bit far away. Since MLM adopts local anchors, these local em-

beddings push contextualized representations into different clusters. The contextual score begins to decrease too. This phenomenon proves the embedding bias problem where the learning of contextualized representations is decided by the selected embeddings where the global contextual semantics are neglected.

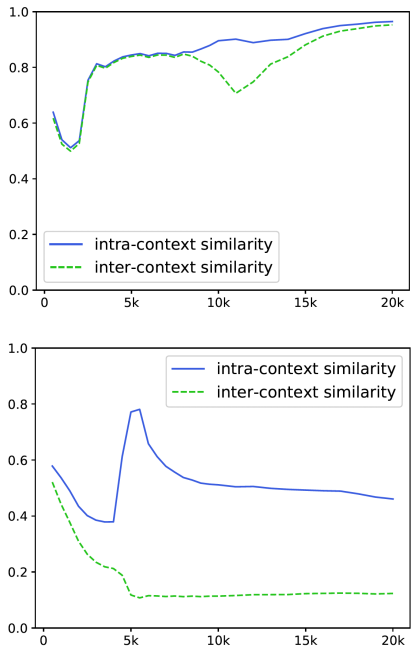

Figure 4: The impact of embedding quality for the learning of contextualized representations. We train two BERT-small variants from scratch, whose embedding is either (a) randomly initialized and frozen or (b) copied from normally pre-trained BERT at 250k steps and frozen.

3 Proposed Approach: TACO

To address the challenges of MLM, we propose a new method TACO to combine global anchors and local anchors. We first introduce TC, a token-alignment contrastive loss which explicitly models global semantics in Section 3.1, and combine TC with MLM to get the overall objective for training our TACO model in Section 3.2.

3.1 Token-alignment Contrastive Loss

To model global semantics, the objective is expected to be capable of explicitly capturing information shared between contextualized representation of tokens within the same context. Therefore, a natural solution is to maximize the mutual information of contextual information hidden in contextualized representations in the same context. To extract shared contextual information, we first define a rule to generate contextual representations of tokens by combining embeddings and global information. Formally,

$$\boldsymbol{h}_i = f(\boldsymbol{e}_i, \boldsymbol{g}). \tag{2}$$

where f is a probing algorithm and $\boldsymbol{e}_i$ is the embedding and $\boldsymbol{g}$ is the global bias of a concrete context. In this paper, we adopt a straightforward probing method to get global information hidden in contextualized representations, where

$$\boldsymbol{g}_i = \boldsymbol{h}_i - \boldsymbol{e}_i. \tag{3}$$

Given contextualized representations of an token $\boldsymbol{x}$ and its nearby tokens $\boldsymbol{c}$ in the same context, we use $\boldsymbol{g}_x$ and $\boldsymbol{g}_c$ to represent global semantics hidden in these representations. The mutual information between the two global bias $\boldsymbol{g}_x$ and $\boldsymbol{g}_c$ is

$$I(\boldsymbol{g}_x, \boldsymbol{g}_c) = \sum_{\boldsymbol{g}_x, \boldsymbol{g}_c} p(\boldsymbol{g}_x, \boldsymbol{g}_c) \log \frac{p(\boldsymbol{g}_x|\boldsymbol{g}_c)}{p(\boldsymbol{g}_x)} \tag{4}$$

According to van den Oord et al. 2019, the InfoNCE loss serves as an estimator of mutual information of $\boldsymbol{x}$ and $\boldsymbol{c}$:

$$I(\boldsymbol{g}_x, \boldsymbol{g}_c) \geq \log(K) - \mathcal{L}(\boldsymbol{g}_x, \boldsymbol{g}_c) \tag{5}$$

where $\mathcal{L}(\boldsymbol{g}_x, \boldsymbol{g}_c)$ is defined as:

$$\mathcal{L}(\boldsymbol{g}_x, \boldsymbol{g}_c) = -\mathbb{E}\left[\log \frac{f(\boldsymbol{g}_x, \boldsymbol{g}_c)}{f(\boldsymbol{g}_x, \boldsymbol{g}_c) + \sum_{k=1}^{K} f(\boldsymbol{g}_x, \boldsymbol{g}_{c_k^-})}\right] \tag{6}$$

where $\boldsymbol{c}_k^-$ is the k-th negative sample of $\boldsymbol{x}$ and K is the size of negative samples. Hence minimizing the objective $\mathcal{L}(\boldsymbol{g}_x, \boldsymbol{g}_c)$ is equivalent to maximizing the lower bound on the mutual information $I(\boldsymbol{g}_x, \boldsymbol{g}_c)$. This objective contains two parts: *positive pairs* $f(\boldsymbol{g}_x, \boldsymbol{g}_c)$ and *negative pairs* $f(\boldsymbol{g}_x, \boldsymbol{g}_{c_k^-})$.

Previous study (Chen et al., 2020) has shown that cosine similarity with temperature performs well as the score function f in InfoNCE loss. Following them, we take

$$f(\boldsymbol{g}_x, \boldsymbol{g}_c) = \frac{1}{\tau} \frac{\boldsymbol{g}_x \cdot \boldsymbol{g}_c}{\|\boldsymbol{g}_x\| \|\boldsymbol{g}_c\|} \tag{7}$$

where τ is the temperature hyper-parameter and $\|\cdot\|$ is ℓ_2-norm function.

Contextualized representation: To get global bias $\boldsymbol{g}_x$ and $\boldsymbol{g}_c$ following Eq. 3, we adopt the widely-used Transformer (Vaswani et al., 2017) as the encoder and take the last hidden states as

the contextualized representations $\boldsymbol{h}_x$ and $\boldsymbol{h}_c$. Formally, suppose a batch of sequences $\{\boldsymbol{s}_i\}$ where $i \in \{1, \cdots, N\}$. We feed it into the Transformer encoder to obtain contextualized representations, $\boldsymbol{h}_1^i, \boldsymbol{h}_2^i, \cdots, \boldsymbol{h}_{|\boldsymbol{s}_i|}^i$ where $\boldsymbol{h}_j^i \in \mathbb{R}^d$.

Positive pairs: Given each token $\boldsymbol{x}$, we randomly sample a positive sample $\boldsymbol{c}$ from nearby tokens in the same context (sequence) within a window span where W is the window size.

Negative pairs: Given each token $\boldsymbol{x}$, we randomly sample K tokens from other sequences in this batch as negative samples $\boldsymbol{c}_k^-$.

To sum up, the **T**oken-alignment **C**ontrastive (**TC**) loss is applied to every token in a batch as:

$$\mathcal{L}_{\text{TC}} = \frac{1}{N} \sum_{i=1}^{N} \frac{1}{|\boldsymbol{s}_i|} \sum_{j=1}^{|\boldsymbol{s}_i|} \mathcal{L}(\boldsymbol{g}_j^i, \boldsymbol{g}_{j_c}^i) \tag{8}$$

where N is the number of sequences of this batch; $\boldsymbol{s}_i$ is the i-th sequence; j and jc are tokens in $\boldsymbol{s}_i$ where $jc \neq j$; $\boldsymbol{g}^i$ is the global semantics hidden in contextualized representation of token $\boldsymbol{s}_i$. $\boldsymbol{g}_j^i$ and $\boldsymbol{g}_{j_c}^i$ are generated via:

$$\boldsymbol{g}_j^i = \boldsymbol{h}_j^i - \boldsymbol{e}_j^i \tag{9}$$

$$\boldsymbol{g}_{j_c}^i = \boldsymbol{h}_{j_c}^i - \boldsymbol{e}_{j_c}^i \tag{10}$$

where $\boldsymbol{h}_j^i$ and $\boldsymbol{e}_j^i$ are the contextualized representation and static embedding of the anchor token, respectively. $\boldsymbol{h}_{j_c}^i$ and $\boldsymbol{e}_{j_c}^i$ are the contextualized representation and static embedding of the sampled positive token in the same context.

3.2 Training Objective

As described before, the token-alignment contrastive loss $\mathcal{L}_{\text{TC}}$ is designed to model global dependencies while MLM is able to capture local dependencies. Therefore, we can better model contextualized representations by combining the token-alignment contrastive loss $\mathcal{L}_{\text{TC}}$ and the MLM loss to get our overall objective $\mathcal{L}_{\text{TACO}}$:

$$\mathcal{L}_{\text{TACO}} = \mathcal{L}_{\text{TC}} + \mathcal{L}_{\text{MLM}} \tag{11}$$

We implement it in a multi-task learning manner where all objectives are calculated within one forward propagation, which only introduces negligible extra computations.

4 Experiments

4.1 Experimental Settings

Training Following BERT (Devlin et al., 2019), we select the BooksCorpus (800M words after WordPiece tokenization) (Zhu et al., 2015) and English Wikipedia (4B words) as pre-training corpus. We pre-train two variants of BERT models: BERT-small and BERT-base. All models are equipped with the vocabulary of size 30,522, trained with 15% masked positions for MLM. The maximum sequence length is 256 and batch size is 1,280. We adopt optimizer AdamW (Loshchilov and Hutter, 2019) with learning rate 1e-4. All models are trained until convergence. To be specific, the small model is trained up to 250k steps with a warm-up of 2.5k steps. The base model is trained up to 500k steps with a warm-up of 10k steps. For TACO, we set the positive sample window size W to 5, the negative sample number K to 50, and the temperature parameter τ to 0.07 after a slight grid-search via preliminary experiments. More pre-training details can be found in Appendix A.

During fine-tuning models, we conduct a grid search over batch sizes of {16, 32, 64, 128}, learning rates of {1e-5, 2e-5, 3e-5, 5e-5}, and training epochs of {4, 6} with an Adam optimizer (Kingma and Ba, 2015). We use the open-source packages for implementation, including HuggingFace Datasets[1] and Transformers[2]. All the experiments are conducted on 16 GPU chips (32 GB V100).

Evaluation We evaluate methods on the GLUE benchmark (Wang et al., 2019). Specifically, we test on Microsoft Research Paraphrase Matching (MRPC) (Dolan and Brockett, 2005), Quora Question Pairs (QQP)[3] and STS-B (Conneau and Kiela, 2018) for Paraphrase Similarity Matching; Stanford Sentiment Treebank (SST-2) (Socher et al., 2013) for Sentiment Classification; Multi-Genre Natural Language Inference Matched (MNLI-m), Multi-Genre Natural Language Inference Mismatched (MNLI-mm) (Williams et al., 2018), Question Natural Language Inference (QNLI) (Rajpurkar et al., 2016) and Recognizing Textual Entailment (RTE) (Wang et al., 2019) for the Natural Language Inference (NLI) task; The Corpus of Linguistic Acceptability (CoLA) (Warstadt et al., 2019) for Linguistic Acceptability.

Following Devlin et al. (2019), we exclude WNLI (Levesque, 2011). We report F1 scores for QQP and MRPC, Spearman correlations for STS-

[1] https://github.com/huggingface/datasets

[2] https://github.com/huggingface/transformers

[3] https://www.quora.com/q/quoradata/First-Quora-Dataset-Release-Question-Pairs

	Approach	MNLI(m/mm)	QQP	QNLI	SST-2	CoLA	STS-B	MRPC	RTE	**Avg.**
Validation Set	MLM-250k	76.9 / 77.4	85.7	86.2	**89.0**	28.8	85.6	85.9	**59.6**	75.0
	TACO-50k	76.7 / 76.8	85.2	85.0	87.5	31.3	85.6	87.1	59.1	74.9
	TACO-250k	**77.9 / 78.4**	**86.1**	**86.5**	88.9	**34.2**	**86.1**	**88.1**	59.5	**76.2**
Test Set	MLM-250k	77.5 / 76.5	**68.2**	85.6	89.3	27.9	76.9	82.6	**60.6**	71.7
	TACO-250k	**78.0 / 76.9**	67.6	**86.3**	**89.5**	**31.2**	**77.8**	**84.4**	58.4	**72.2**

Table 1: GLUE results on BERT-small. For validation results, we run 4 experiments with different seeds for each task and report the average score. For test results, we report the test scores of the checkpoint performing best on validation sets. TACO outperforms MLM with 1.2 point improvement and 5× speedup on validations sets. On test sets, TACO also obtains better results on 6 out of 8 tasks.

B, and accuracy scores for the other tasks. For evaluation results on validation sets, we report the average score of 4 fine-tunings with different random seeds. For results on test sets, we select the best model on the validation set to evaluate.

Baselines We mainly compare TACO with MLM on BERT-small and BERT-base models. In addition, we also compare TACO with related contrastive methods: a sentence-level contrastive method BERT-NCE and a span-based contrastive learning method INFOWORD, both from Kong et al. (2020). We directly compare TACO with the results reported in their paper.

4.2 Results on BERT-Small

Table 1 and Figure 5 show the results of TACO on BERT-small. As we can see, compared with MLM with 250k training steps (convergence steps), TACO achieves comparable performance with only 1/5 computation budget. By modeling global dependencies, TACO can significantly improve the efficiency of contextualized representation learning. In addition, when pre-trained with the same steps, TACO outperforms MLM with 1.2 average score improvement on the validation set.

In addition to convergence, we also compare TACO and MLM on fewer training data. The results are shown in Table 2. We sample 4 tasks with the largest amount of training data for evaluation. As we can see, TACO trained on 25% data can achieve competitive results with MLM trained on full data. These results also verify the data efficiency of our method, TACO.

4.3 Results on BERT-Base

We also compare TACO with MLM on base-sized models, which are the most commonly used models according to the download data from Huggingface[4] (Wolf et al., 2020). First, from Table 3, we can see that TACO consistently outperforms MLM under all pre-training computation budgets. Notably, TACO-250k achieves comparable performance with MLM-500k, which saves 2x computations. Similar results are observed on TACO-100k and BERT-250k. These results demonstrate that TACO can achieve better acceleration over MLM. It is also a significant improvement compared to previous methods (Gong et al., 2019) focusing on accelerating BERT but only with slight speedups.

Approach	MNLI	QQP	QNLI	SST-2	**Avg.**
MLM-25%	77.8	85.7	85.8	87.2	84.1
MLM-100%	76.9	85.7	86.2	**89.0**	84.5
TACO-25%	77.8	85.7	86.1	88.4	84.5
TACO-100%	**77.9**	**86.1**	**86.5**	88.9	**84.9**

Table 2: TACO pre-trained on a quarter of data achieves competitive downstream results with MLM pre-trained on full data. All results are reported on GLUE validation sets with BERT-small. Here we sample 4 tasks with the largest amount of training data.

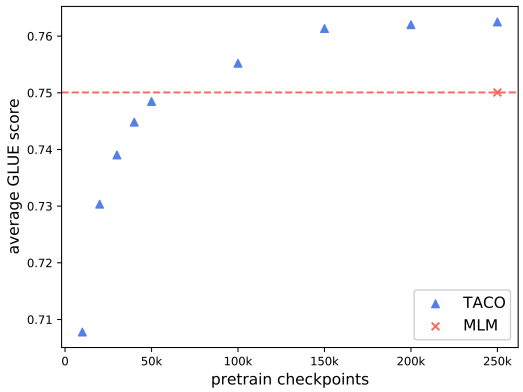

Figure 5: Average GLUE score during pre-training. All results are reported on validation sets with BERT-small. TACO achieves better results and 5× speedup than MLM.

[4] https://huggingface.co/models

Approach	MNLI	QQP	QNLI	SST-2	CoLA	STS-B	MRPC	RTE	**Avg.**
MLM-100k	80.7	86.4	89.3	90.5	47.4	86.0	85.0	56.6	77.7
MLM-250k	83.0	87.4	90.4	91.8	48.6	87.1	87.5	57.8	79.2
MLM-500k	84.2	87.9	**91.1**	92.1	51.1	87.9	89.8	63.4	80.9
TACO-100k	81.5	87.4	89.4	90.3	46.4	87.2	87.8	62.8	79.1
TACO-250k	83.8	87.9	90.2	91.4	50.7	87.9	89.3	63.5	80.6
TACO-500k	**84.6**	**88.1**	90.8	**92.3**	**53.4**	**88.5**	**90.7**	**66.3**	**81.8**

Table 3: GLUE results on BERT-base. All results are reported on validation sets. We run 6 experiments with different hyper-parameter combinations (including random seeds) for each task and report the average score. The MNLI-matched score is reported here. TACO outperforms MLM with 0.9 point improvement and 2× speedup.

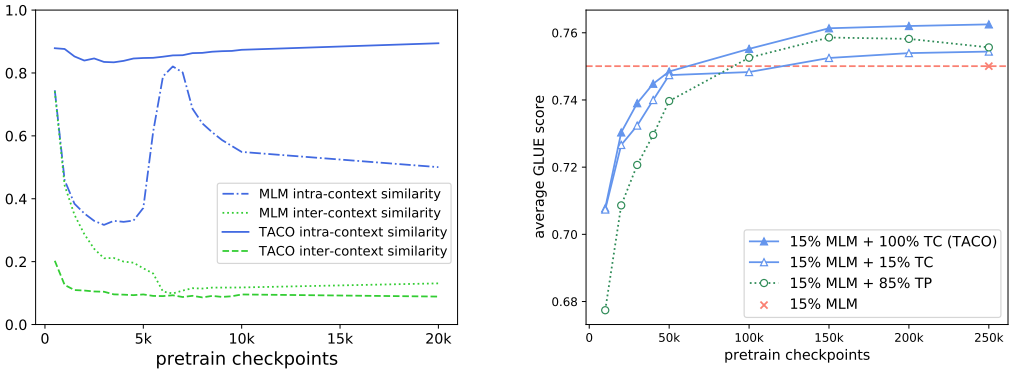

Figure 6: The left figure (a) shows the intra-context similarity and inter-context similarity change during pre-training. The right figure (b) shows two ablations of TACO: a concentrated TACO (15% MLM + 15% TC), where the MLM loss and the TC loss are both built on the same 15% masked positions, and an extended MLM (15% MLM + 85% TP), which masks 15% positions but predict original tokens on all positions.

In addition, as shown in Table 4, TACO achieves competitive results compared to BERT-NCE and INFOWORD, two similar contrastive methods.

5 Discussion

5.1 TACO and MLM

To better understand how TACO works, we conduct a quantitative comparison on the learning dynamic for BERT and TACO. Similar to Section 2.2, we plot the Cosine similarity among contextualized representations of tokens in the same context (intra-context) and different contexts (inter-context) in Figure 6. We find that the learning dynamic of TACO significantly differs from that of MLM. Specifically, for TACO, the intra-context representation similarity remains high and the gap between intra-context similarity and inter-context similarity remains large at the later stage of training. This confirms that TACO can better fulfill global semantics, which may contribute to the superior downstream performance.

5.2 Ablation Study

TACO is implemented as a token-level contrastive (TC) loss along with the MLM loss. Therefore, the improvement of TACO might come from two aspects, including 1) denser supervision signals from the all-token objective and 2) the benefits of the contrastive loss to strengthen global dependencies. It is helpful to figure out which factor is more important. To this end, we design two variants for ablation. One is a *concentrated* TACO, where the contrastive loss is built on the 15% masked positions only, keeping the same density of supervision signal with MLM. The other is an *extended* MLM, where not only 15% masked positions are asked to predict the original token, so do the rest 85% unmasked positions. The extended MLM has the same dense supervision with TACO but loses the benefits of modeling the global dependencies. The results on small models are shown in Figure 6.

As we can see, the performance of TACO decreases if we sample a part of token positions to implement TC objectives. It shows that more supervision signals benefit the final performance of

Approach	MNLI(m/mm)	QQP	QNLI	SST-2	**Avg.**
BERT-NCE	83.2 / 83.0	70.5	90.9	93.0	84.1
INFOWORD	83.7 / 82.4	71.0	91.4	92.5	84.2
TACO	**84.5 / 83.5**	**71.7**	**91.6**	**93.2**	**84.9**

Table 4: TACO achieves the best among contrastive-based methods. All results are reported on GLUE test sets with BERT-base. For each task, we report test results of the checkpoint performing best on validation sets.

TACO. However, simply adding more supervision signals by predicting unmasked tokens does not help MLM too much. Even equipped with the extra 85% token prediction (TP) loss, MLM+TP does not show significant improvements and it is noticeable that the performance of MLM+TP starts to drop after 150k steps. This further confirms the effectiveness of TC loss by strengthening global dependencies.

6 Related Work

6.1 Language Representation Learning

Classic language representation learning methods (Mikolov et al., 2013; Pennington et al., 2014) aims to learn context-independent representation of words, i.e., word embeddings. They generally follow the distributional hypothesis (Harris, 1954). Recently, the pre-training then fine-tuning paradigm has become a common practice in NLP because of the success of pre-trained language models like BERT (Devlin et al., 2019). Context-dependent (or contextualized) representations are the basic characteristic of these methods. Many existing contextualized models are based on the masked language modeling objective, which randomly masks a portion of tokens in a text sequence and trains the model to recover the masked tokens. Many previous studies prove that pre-training with the MLM objective helps the models learn syntactical and semantic knowledge (Clark et al., 2019). There have been numerous extensions to MLM. For example, XLNet (Yang et al., 2019) introduced the permutated language modeling objective, which predicts the words one by one in a permutated order. BART (Lewis et al., 2020) and T5 (Raffel et al., 2020) investigated several denoising objectives and pre-trained an encoder-decoder architecture with the mask span infilling objective. In this work, we focus on the key MLM objective and aim to explore how MLM objective helps learn contextualized representation.

6.2 Contrastive-based SSL

Apart from denoising-based objectives, contrastive learning is another promising way to obtain self-supervision. In contrastive-based self-supervised learning, the models are asked to distinguish the positive samples from the negative ones for a given anchor. Contrastive-based SSL method was first introduced in NLP for efficient learning of word representations by negative sampling, i.e., SGNS (Word2Vec (Mikolov et al., 2013)). Later, similar ideas were brought into CV field for learning image representation and got prevalent, such as MoCo (He et al., 2020), SimCLR (Chen et al., 2020), BYOL (Caron et al., 2020), etc.

In the recent two years, there have been many studies targeting at reviving contrastive learning for contextual representation learning in NLP. For instance, CERT (Fang et al., 2020) utilized back-translation to generate positive pairs. CAPT (Luo et al., 2020) applied masks to the original sentence and considered the masked sentence and its original version as the positive pair. DeCLUTR (Giorgi et al., 2020) samples nearby even overlapping spans as positive pairs. INFOWORD (Kong et al., 2020) treated two complementary parts of a sentence as the positive pair. However, the aforementioned methods mainly focus on sentence-level or span-level contrast and may not provide dense self-supervision to improve efficiency. Unlike these approaches, TACO regards the global semantics hidden in contextualized token representations as the positive pair. The token-level contrastive loss can be built on all input tokens, which provides a dense self-supervised signal.

Another related work is ELECTRA (Clark et al., 2020). ELECTRA samples machine-generated tokens from a separate generator and trains the main model to discriminate between machine-generated tokens and original tokens. ELECTRA implicitly treats the fake tokens as negative samples of the context, and the unchanged tokens as positive samples. Unlike this method, TACO does not require architectural modifications and can serve as a plug-

and-play auxiliary objective, largely improving pre-training efficiency.

7 Conclusion

In this paper, we propose a simple yet effective objective to learn contextualized representation. Taking MLM as an example, we investigate whether and how current language model pre-training objectives learn contextualized representation. We find that the MLM objective mainly focuses on local anchors to align contextualized representations, which harms global dependencies modeling due to an "embedding bias" problem. Motivated by these problems, we propose TACO to directly model global semantics. It can be easily combined with existing LM objectives. By combining local and global anchors, TACO achieves up to 5× speedups and up to 1.2 improvements on GLUE score. This demonstrates the potential of TACO to serve as a plug-and-play approach to improve contextualized representation learning.

Acknowledgement

We thank the anonymous reviewers for their helpful feedback. We also thank the colleagues from ByteDance AI Lab for their suggestions on our experiment designing and paper writing.

References

Eneko Agirre, Enrique Alfonseca, Keith Hall, Jana Kravalova, Marius Pasca, and Aitor Soroa. 2009. A study on similarity and relatedness using distributional and wordnet-based approaches.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, Christopher Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*.

Mathilde Caron, Ishan Misra, Julien Mairal, Priya Goyal, Piotr Bojanowski, and Armand Joulin. 2020. Unsupervised learning of visual features by contrasting cluster assignments. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey E. Hinton. 2020. A simple framework for contrastive learning of visual representations. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, pages 1597–1607. PMLR.

Kevin Clark, Urvashi Khandelwal, Omer Levy, and Christopher D. Manning. 2019. What does BERT look at? an analysis of bert's attention. *CoRR*, abs/1906.04341.

Kevin Clark, Minh-Thang Luong, Quoc V. Le, and Christopher D. Manning. 2020. ELECTRA: pre-training text encoders as discriminators rather than generators. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

Alexis Conneau and Douwe Kiela. 2018. Senteval: An evaluation toolkit for universal sentence representations. In *LREC*.

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, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 4171–4186. Association for Computational Linguistics.

William B. Dolan and Chris Brockett. 2005. Automatically constructing a corpus of sentential paraphrases. In *IWP@IJCNLP*.

Hongchao Fang, Sicheng Wang, Meng Zhou, Jiayuan Ding, and Pengtao Xie. 2020. Cert: Contrastive self-supervised learning for language understanding. *arXiv preprint arXiv:2005.12766*.

John M Giorgi, Osvald Nitski, Gary D Bader, and Bo Wang. 2020. Declutr: Deep contrastive learning for unsupervised textual representations. *arXiv preprint arXiv:2006.03659*.

Linyuan Gong, Di He, Zhuohan Li, Tao Qin, Liwei Wang, and Tie-Yan Liu. 2019. Efficient training of BERT by progressively stacking. In *ICML*, volume 97 of *Proceedings of Machine Learning Research*, pages 2337–2346. PMLR.

Zellig S Harris. 1954. Distributional structure. *Word*, 10(2-3):146–162.

Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross B. Girshick. 2020. Momentum contrast for unsupervised visual representation learning. In *2020*

IEEE/CVF Conference on Computer Vision and Pattern Recognition, CVPR 2020, Seattle, WA, USA, June 13-19, 2020, pages 9726–9735. Computer Vision Foundation / IEEE.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Lingpeng Kong, Cyprien de Masson d'Autume, Lei Yu, Wang Ling, Zihang Dai, and Dani Yogatama. 2020. A mutual information maximization perspective of language representation learning. In *International Conference on Learning Representations*.

Hector J. Levesque. 2011. The winograd schema challenge. In *AAAI Spring Symposium: Logical Formalizations of Commonsense Reasoning*.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 7871–7880. 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. 2019. Roberta: A robustly optimized BERT pretraining approach. *CoRR*, abs/1907.11692.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net.

Fuli Luo, Pengcheng Yang, Shicheng Li, Xuancheng Ren, and Xu Sun. 2020. Capt: Contrastive pre-training for learningdenoised sequence representations. *arXiv preprint arXiv:2010.06351*.

Tomás Mikolov, Ilya Sutskever, Kai Chen, Gregory S. Corrado, and Jeffrey Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems 26: 27th Annual Conference on Neural Information Processing Systems 2013. Proceedings of a meeting held December 5-8, 2013, Lake Tahoe, Nevada, United States*, pages 3111–3119.

Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP 2014, October 25-29, 2014, Doha, Qatar, A meeting of SIGDAT, a Special Interest Group of the ACL*, pages 1532–1543. ACL.

Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2018, New Orleans, Louisiana, USA, June 1-6, 2018, Volume 1 (Long Papers)*, pages 2227–2237. Association for Computational Linguistics.

Xipeng Qiu, Tianxiang Sun, Yige Xu, Yunfan Shao, Ning Dai, and Xuanjing Huang. 2020. Pre-trained models for natural language processing: A survey. *CoRR*, abs/2003.08271.

Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. Improving language understanding by generative pre-training.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.*, 21:140:1–140:67.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. Squad: 100, 000+ questions for machine comprehension of text. In *EMNLP*.

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

Aaron van den Oord, Yazhe Li, and Oriol Vinyals. 2019. Representation learning with contrastive predictive coding.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 5998–6008.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. 2019. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *ICLR*.

Tongzhou Wang and Phillip Isola. 2020. Understanding contrastive representation learning through alignment and uniformity on the hypersphere. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, pages 9929–9939. PMLR.

Alex Warstadt, Amanpreet Singh, and Samuel R. Bowman. 2019. Neural network acceptability judgments. *TACL*.

Adina Williams, Nikita Nangia, and Samuel R. Bowman. 2018. A broad-coverage challenge corpus for sentence understanding through inference. In *NAACL-HLT*.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations, EMNLP 2020 - Demos, Online, November 16-20, 2020*, pages 38–45. Association for Computational Linguistics.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime G. Carbonell, Ruslan Salakhutdinov, and Quoc V. Le. 2019. Xlnet: Generalized autoregressive pretraining for language understanding. In *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, pages 5754–5764.

Yukun Zhu, Ryan Kiros, Rich Zemel, Ruslan Salakhutdinov, Raquel Urtasun, Antonio Torralba, and Sanja Fidler. 2015. Aligning books and movies: Towards story-like visual explanations by watching movies and reading books. In *Proceedings of the IEEE international conference on computer vision*, pages 19–27.

A Experiment Details

A.1 Pre-training Hyper-parameters

All pre-training approaches involved in experiments use the same pre-training hyper-parameters but do not include BERT-NCE and INFOWORD. Results of BERT-NCE and INFOWORD are directly cited from the original paper (Kong et al., 2020). Following Liu et al. (2019), we do not use the next sentence prediction (NSP) objective and use dynamic masking for MLM with a 15% mask ratio, where the masked positions are decided on the fly.

TACO introduces three extra hyper-parameters, including negative sample size K, positive sample window size W and temperature τ. We set the temperature τ as a small value, 0.07, following Fang et al. (2020). By searching for the best K out of $\{10, 50\}$ and W out of $\{3, 5, 10, 50\}$ on the small TACO model, we found that TACO with K=50 and W=5 performs best, so we also apply these hyper-parameter choices for base-sized TACO. The full set of pre-training hyper-parameters are listed in Table 5. Actually, TACO outperforms MLM under most cases in our preliminary experiments. However, we still also find some extreme cases which might harm the effectiveness of TACO. If the size of negative samples K is too small, e.g., smaller than 10, the performance of TACO degenerates nearly to the level of BERT baseline. Similar conclusions are also mentioned in related works (He et al., 2020; Chen et al., 2020). Also, if the positive window size W is too large, e.g., bigger than 50, the performance of TACO degrades, too. We suspect the over-large positive window brings more false-positive samples, which makes the sequence meaning ambiguous, thus harms the performance.

A.2 Fine-tuning Details

For small-sized models, we fine-tune all saved checkpoints (5k, 10k, 20k, 30k, 40k, 50k, 100k, 150k, 200k, 250k-step) of different pre-trained models (TACO and its ablations) with the same hyper-parameters on each task. Considering the large amount of pre-training checkpoints, we just adopt the default fine-tuning hyper-parameters and repeat fine-tuning 4 times with different random seeds. Then the best performed fine-tuned models on validation sets are used for testing. This setting helps make a fair comparison among models and avoids a large amount of grid-search runs. The task-specific hyper-parameters for small-sized models

Pre-training	Hyper-parameters	Small	Base
Parameters Shared by All Approaches	Number of Layers	4	12
	Hidden Size	512	768
	Hidden Layer Activation Function	gelu	gelu
	FFN Inner Hidden Size	2,048	3,072
	Attention Heads	8	12
	Attention Head Size	64	64
	Embedding Size	512	768
	Vocab Size	30,522	30,522
	Max Position Embeddings	512	512
	Max Sequence Length	256	256
	Attention Dropout	0.1	0.1
	Dropout	0.1	0.1
	Initializer Range	0.02	0.02
	Learning Rate Decay	Linear	Linear
	Learning Rate	1e-4	1e-4
	Max Gradient Norm	1.0	1.0
	Adam ϵ	1e-8	1e-8
	Adam β_1	0.9	0.9
	Adam β_2	0.999	0.999
	Weight Decay	0.01	0.01
	Batch Size	1,280	1,280
	Train Steps	250k	500k
	Warm-up Steps	2,500	10,000
	FP16	True	True
	Mask Percentage	15	15
TACO Only	Negative Sample Size K	50	50
	Positive Sample Window Size W	5	5
	Temperature Parameter τ	0.07	0.07

Table 5: Hyper-parameters during pre-training.

Fine-tuning	Hyper-parameters	Small/Base
Parameters Shared by All Models	Max Sequence Length	128
	Attention Dropout	0.1
	Dropout	0.1
	Initializer Range	0.02
	Learning Rate Decay	Linear
	Max Gradient Norm	1.0
	Adam ϵ	1e-8
	Adam β_1	0.9
	Adam β_2	0.999
	Weight Decay	0.0
	FP16	False

Table 6: Hyper-parameters during fine-tuning.

Task	Learning Rate	Batch Size	Train Epochs	Warm-up Steps
MNLI	5e-5	64	6	2,000
QQP	5e-5	64	6	2,000
QNLI	5e-5	64	4	200
SST-2	5e-5	64	4	200
CoLA	5e-5	32	4	100
STS-B	5e-5	32	4	100
MRPC	5e-5	32	4	100
RTE	5e-5	32	4	100

Table 7: Task-specific hyper-parameters for small models during fine-tuning.

are listed in Table 7. The general fine-tuning hyper-parameters are listed in Table 6.

For base-sized models, we save checkpoints at 100k, 250k, and 500k steps, respectively. During fine-tuning, we also conduct multiple fine-tuning runs with different task-specific hyper-parameter combinations as shown in Table 8. Concretely, we randomly sample 6 different hyper-parameter combinations and report the average score for validation results. Then we select the best-performing run of 500k-step checkpoints (converged) for testing.

A.3 Statistic Details

Embedding Similarity We calculate cosine similarity of 20 randomly sampled pairs of frequently co-occurrent words from the WordSim353 dataset (Agirre et al., 2009) labeled by human annotators to plot the average similarity curve in Figure 3(b). Corresponding embeddings are obtained from the embedding layer of the BERT model and variant models mentioned in Section 2.2.

Intra-/Inter-context Similarity For every token w_i in the corpus, we randomly sample a positive token $w_{j\neq i}$ within the same context (sentence) and another token w_k from other sentences. As mentioned in Section 2.2, we take BERT (Devlin et al., 2019) as our encoder to get contextualized representations through the last hidden states $\boldsymbol{h}$. We mainly adopt the cosine similarity as the measurement and calculate the average intra-context similarity (between $\boldsymbol{h}_i$ and $\boldsymbol{h}_j$) and the average inter-context similarity (between $\boldsymbol{h}_i$ and $\boldsymbol{h}_k$) over all tokens in the corpus. It is worth noticing that we do use any masks here when generating a token's contextualized representation for statistics.

Other Measurements We observe the same findings for MLM under other measurements, though the statistics before are mainly based on cosine similarities. We tried other similarities or distances, e.g., L1 distance, L2 distance and L10 distance, to evaluate the discrepancy between contextualized representations from the same context and different contexts. Specifically, we make intra-context and inter-context statistics under specific measurement at different pre-training checkpoints, then calculate the ratio of intra-context measurement over the inter-context one. Table 9 shows the statistical results. As we can see, when the ratio of L1 distance decreases, the ratio of cosine similarity and the dot-production similarity increase, vice versa.

B Extra Experiments

In the standard implementation of BERT, the parameters of input embeddings are shared with output embeddings. All experiments and analyses in this paper are based on this assumption. To further

Task	Learning Rate	Batch Size	Train Epochs	Warm-up Steps
MNLI	{1e-5, 2e-5, 3e-5, 5e-5}	{32, 64, 128}	{4, 6, 8}	{1000, 2000}
QQP	{1e-5, 2e-5, 3e-5, 5e-5}	{32, 64, 128}	{4, 6, 8}	{1000, 2000}
QNLI	{1e-5, 2e-5, 3e-5, 5e-5}	{32, 64}	{4, 6}	{100, 200, 1000}
SST-2	{1e-5, 2e-5, 3e-5, 5e-5}	{16, 32, 64}	{4, 6}	200
CoLA	{1e-5, 2e-5, 3e-5, 5e-5}	{16, 32, 64}	{4, 6}	100
STS-B	{1e-5, 2e-5, 3e-5, 5e-5}	{16, 32, 64}	{4, 6}	100
MRPC	{1e-5, 2e-5, 3e-5, 5e-5}	{16, 32, 64}	{4, 6}	100
RTE	{1e-5, 2e-5, 3e-5, 5e-5}	{16, 32, 64}	{4, 6, 8}	100

Table 8: Task-specific hyper-parameters for base models during fine-tuning.

Measurement / Checkpoint	1k	2k	3k	5k	7.5k	10k	20k	50k	100k	250k
L1 Distance	0.977	0.925	0.880	0.833	0.769	0.779	0.774	0.797	0.820	0.838
L2 Distance	0.978	0.927	0.884	0.838	0.778	0.789	0.783	0.803	0.826	0.843
L10 Distance	0.981	0.928	0.890	0.854	0.802	0.811	0.805	0.822	0.844	0.860
Cosine Similarity	1.093	1.314	1.548	1.890	3.197	3.533	3.591	3.482	3.325	3.174
Dot-production Similarity	1.092	1.313	1.547	1.890	3.189	3.525	3.586	3.480	3.321	3.166

Table 9: The ratio of intra-context measurement over inter-context measurement during pre-training. We list two distance measurements and three similarity measurements here.

confirm the effectiveness of TACO, we conduct the extra experiments without embedding sharing on BERT-small. The results are showed in Table 10. It is unexpected that the variants without embedding sharing perform worse compared their counterparts due to lack of regularization of weight sharing. From the results, we can see that the TACO without embedding sharing performs slightly worse than TACO with embedding sharing. However, compared to the MLM, it is still better than MLM than 0.9 average GLUE score when convergence. These results prove the effectiveness of TACO even when embeddings are not sharing.

Approach	MNLI	QQP	QNLI	SST-2	CoLA	STS-B	MRPC	RTE	**Avg.**
MLM-250k	76.9 / 77.4	85.7	86.2	**89.0**	28.8	85.6	85.9	**59.6**	75.0
TACO-50k	76.7 / 76.8	85.2	85.0	87.5	31.3	85.6	87.1	59.1	74.9
TACO-50k w/o shared embedding	76.3 / 76.5	85.0	85.2	87.2	32.5	85.1	86.7	58.9	74.6
TACO-250k	**77.9 / 78.4**	86.1	**86.5**	88.9	34.2	**86.1**	**88.1**	59.5	**76.2**
TACO-250k w/o shared embedding	77.5 / 78.2	**86.3**	86.2	88.5	**35.1**	85.8	88.0	59.3	75.9

Table 10: Results on GLUE validation set with small-size models. For models without embedding sharing, we run 3 experiments with different random seeds for each task and report the average score.