ERNIE-Gram: Pre-Training with Explicitly N-Gram Masked Language Modeling for Natural Language Understanding

Dongling Xiao, Yu-Kun Li, Han Zhang, Yu Sun, Hao Tian, Hua Wu and **Haifeng Wang**
Baidu Inc., China
{xiaodongling,liyukun01,zhanghan17,sunyu02,
tianhao,wu_hua,wanghaifeng}@baidu.com

Abstract

Coarse-grained linguistic information, such as named entities or phrases, facilitates adequately representation learning in pre-training. Previous works mainly focus on extending the objective of BERT's Masked Language Modeling (MLM) from masking individual tokens to contiguous sequences of n tokens. We argue that such contiguously masking method neglects to model the intra-dependencies and inter-relation of coarse-grained linguistic information. As an alternative, we propose ERNIE-Gram, an explicitly n-gram masking method to enhance the integration of coarse-grained information into pre-training. In ERNIE-Gram, n-grams are masked and predicted directly using explicit n-gram identities rather than contiguous sequences of n tokens. Furthermore, ERNIE-Gram employs a generator model to sample plausible n-gram identities as optional n-gram masks and predict them in both coarse-grained and fine-grained manners to enable comprehensive n-gram prediction and relation modeling. We pre-train ERNIE-Gram on English and Chinese text corpora and fine-tune on 19 downstream tasks. Experimental results show that ERNIE-Gram outperforms previous pre-training models like XLNet and RoBERTa by a large margin, and achieves comparable results with state-of-the-art methods. The source codes and pre-trained models have been released at `https://github.com/PaddlePaddle/ERNIE`.

1 Introduction

Pre-trained on large-scaled text corpora and fine-tuned on downstream tasks, self-supervised representation models (Radford et al., 2018; Devlin et al., 2019; Liu et al., 2019; Yang et al., 2019; Lan et al., 2020; Clark et al., 2020) have achieved remarkable improvements in natural language understanding (NLU). As one of the most prominent pre-trained models, BERT (Devlin et al., 2019) employs masked language modeling (MLM) to learn representations by masking individual tokens and predicting them based on their bidirectional context. However, BERT's MLM focuses on the representations of fine-grained text units (e.g. words or subwords in English and characters in Chinese), rarely considering the coarse-grained linguistic information (e.g. named entities or phrases in English and words in Chinese) thus incurring inadequate representation learning.

Many efforts have been devoted to integrate coarse-grained semantic information by independently masking and predicting contiguous sequences of n tokens, namely n-grams, such as named entities, phrases (Sun et al., 2019b), whole words (Cui et al., 2019) and text spans (Joshi et al., 2020). We argue that such contiguously masking strategies are less effective and reliable since the prediction of tokens in masked n-grams are independent of each other, which neglects the intra-dependencies of n-grams. Specifically, given a masked n-gram $\boldsymbol{w} = \{x_1, ..., x_n\}, x \in \mathcal{V}_F$, we maximize $p(\boldsymbol{w}) = \prod_{i=1}^{n} p(x_i|\boldsymbol{c})$ for n-gram learning, where models learn to recover $\boldsymbol{w}$ in a huge and sparse prediction space $\mathcal{F} \in \mathbb{R}^{|\mathcal{V}_F|^n}$. Note that $\mathcal{V}_F$ is the fine-grained vocabulary[1] and $\boldsymbol{c}$ is the context.

We propose ERNIE-Gram, an **explicitly n-gram masked** language modeling method in which n-grams are masked with single `[MASK]` symbols, and predicted directly using explicit n-gram identities rather than sequences of tokens, as depicted in Figure 1(b). The models learn to predict n-gram $\boldsymbol{w}$ in a small and dense prediction space $\mathcal{N} \in \mathbb{R}^{|\mathcal{V}_N|}$, where $\mathcal{V}_N$ indicates a prior n-gram lexicon[2] and normally $|\mathcal{V}_N| \ll |\mathcal{V}_F|^n$. To

[1] $\mathcal{V}_F$ contains 30K BPE codes in BERT (Devlin et al., 2019) and 50K subword units in RoBERTa (Liu et al., 2019).

[2] $\mathcal{V}_N$ contains 300K n-grams, where $n \in [2, 4)$ in this paper, n-grams are extracted in word-level before tokenization.

Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pages 1702–1715
June 6–11, 2021. ©2021 Association for Computational Linguistics

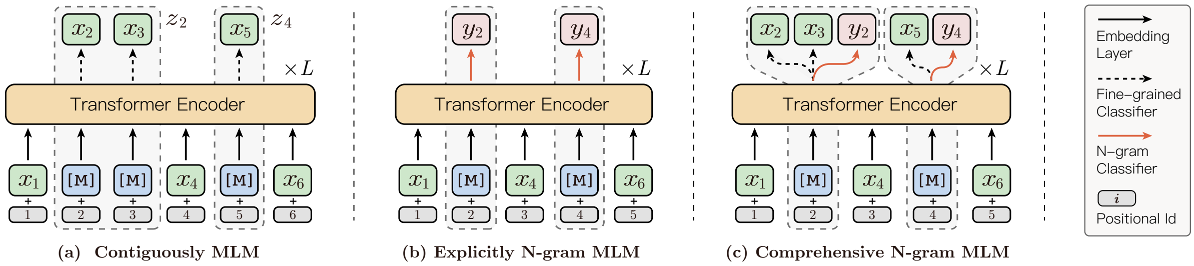

Figure 1: Illustrations of different MLM objectives, where x_i and y_i represent the identities of fine-grained tokens and explicit n-grams respectively. Note that the weights of fine-grained classifier ($W_F \in \mathbb{R}^{h\times|\mathcal{V}_F|}$) and N-gram classifier ($W_N \in \mathbb{R}^{h\times|\langle\mathcal{V}_F,\mathcal{V}_N\rangle|}$) are not used in fine-tuning stage, where h is the hidden size and L is the layers.

learn the semantic of n-grams more adequately, we adopt a **comprehensive n-gram prediction** mechanism, simultaneously predicting masked n-grams in coarse-grained (explicit n-gram identities) and fine-grained (contained token identities) manners with well-designed attention mask metrics, as shown in Figure 1(c).

In addition, to model the semantic relationships between n-grams directly, we introduce an **enhanced n-gram relation modeling** mechanism, masking n-grams with plausible n-grams identities sampled from a generator model, and then recovering them to the original n-grams with the pair relation between plausible and original n-grams. Inspired by ELECTRA (Clark et al., 2020), we incorporate the replaced token detection objective to distinguish original n-grams from plausible ones, which enhances the interactions between explicit n-grams and fine-grained contextual tokens.

In this paper, we pre-train ERNIE-Gram on both base-scale and large-scale text corpora (16GB and 160GB respectively) under comparable pre-training setting. Then we fine-tune ERNIE-Gram on 13 English NLU tasks and 6 Chinese NLU tasks. Experimental results show that ERNIE-Gram consistently outperforms previous well-performed pre-training models on various benchmarks by a large margin.

2 Related Work

2.1 Self-Supervised Pre-Training for NLU

Self-supervised pre-training has been used to learn contextualized sentence representations though various training objectives. GPT (Radford et al., 2018) employs unidirectional language modeling (LM) to exploit large-scale corpora. BERT (Devlin et al., 2019) proposes masked language modeling (MLM) to learn bidirectional representations efficiently, which is a representative objective for pre-training and has numerous extensions such as RoBERTa (Liu et al., 2019), UniLM (Dong et al., 2019) and ALBERT (Lan et al., 2020). XL-Net (Yang et al., 2019) adopts permutation language modeling (PLM) to model the dependencies among predicted tokens. ELECTRA introduces replaced token detection (RTD) objective to learn all tokens for more compute-efficient pre-training.

2.2 Coarse-grained Linguistic Information Incorporating for Pre-Training

Coarse-grained linguistic information is indispensable for adequate representation learning. There are lots of studies that implicitly integrate coarse-grained information by extending BERT's MLM to contiguously masking and predicting contiguous sequences of tokens. For example, ERNIE (Sun et al., 2019b) masks named entities and phrases to enhance contextual representations, BERT-wwm (Cui et al., 2019) masks whole Chinese words to achieve better Chinese representations, SpanBERT (Joshi et al., 2020) masks contiguous spans to improve the performance on span selection tasks.

A few studies attempt to inject the coarse-grained n-gram representations into fine-grained contextualized representations explicitly, such as ZEN (Diao et al., 2020) and AMBERT (Zhang and Li, 2020), in which additional transformer encoders and computations for explicit n-gram representations are incorporated into both pre-training and fine-tuning. Li et al., 2019 demonstrate that explicit n-gram representations are not sufficiently reliable for NLP tasks because of n-gram data sparsity and the ubiquity of out-of-vocabulary n-grams. Differently, we only incorporate n-gram information by leveraging auxiliary n-gram classifier and embedding weights in pre-training, which will be completely removed during fine-tuning, so our method maintains the same parameters and computations as BERT.

3 Proposed Method

In this section, we present the detailed implementation of ERNIE-Gram, including n-gram lexicon

$\mathcal{V}_N$ extraction in Section 3.5, explicitly n-gram MLM pre-training objective in Section 3.2, comprehensive n-gram prediction and relation modeling mechanisms in Section 3.3 and 3.4.

3.1 Background

To inject n-gram information into pre-training, many works (Sun et al., 2019b; Cui et al., 2019; Joshi et al., 2020) extend BERT's masked language modeling (MLM) from masking individual tokens to contiguous sequences of n tokens.

Contiguously MLM. Given input sequence $\boldsymbol{x} = \{x_1, ..., x_{|\boldsymbol{x}|}\}, x \in \mathcal{V}_F$ and n-gram starting boundaries $\boldsymbol{b} = \{b_1, ..., b_{|\boldsymbol{b}|}\}$, let $\boldsymbol{z} = \{z_1, ..., z_{|\boldsymbol{b}|-1}\}$ to be the sequence of n-grams, where $z_i = \boldsymbol{x}_{[b_i:b_{i+1})}$, MLM samples 15% of starting boundaries from $\boldsymbol{b}$ to mask n-grams, donating $\mathcal{M}$ as the indexes of sampled starting boundaries, $\boldsymbol{z}_{\mathcal{M}}$ as the contiguously masked tokens, $\boldsymbol{z}_{\setminus\mathcal{M}}$ as the sequence after masking. As shown in Figure 1(a), $\boldsymbol{b} = \{1, 2, 4, 5, 6, 7\}$, $\boldsymbol{z} = \{x_1, \boldsymbol{x}_{[2:4)}, x_4, x_5, x_6\}$, $\mathcal{M} = \{2, 4\}$, $\boldsymbol{z}_{\mathcal{M}} = \{\boldsymbol{x}_{[2:4)}, x_5\}$, and $\boldsymbol{z}_{\setminus\mathcal{M}} = \{x_1, \texttt{[M]}, \texttt{[M]}, x_4, \texttt{[M]}, x_6\}$. Contiguously MLM is performed by minimizing the negative likelihood:

$$-\log p_\theta(\boldsymbol{z}_{\mathcal{M}}|\boldsymbol{z}_{\setminus\mathcal{M}}) = -\sum_{z \in \boldsymbol{z}_{\mathcal{M}}} \sum_{x \in z} \log p_\theta(x|\boldsymbol{z}_{\setminus\mathcal{M}}). \quad (1)$$

3.2 Explicitly N-gram Masked Language Modeling

Different from contiguously MLM, we employ explicit n-gram identities as pre-training targets to reduce the prediction space for n-grams. To be specific, let $\boldsymbol{y} = \{y_1, ..., y_{|\boldsymbol{b}|-1}\}, y \in \langle \mathcal{V}_F, \mathcal{V}_N \rangle$ to be the sequence of explicit n-gram identities, $\boldsymbol{y}_{\mathcal{M}}$ to be the target n-gram identities, and $\bar{\boldsymbol{z}}_{\setminus\mathcal{M}}$ to be the sequence after explicitly masking n-grams. As shown in Figure 1(b), $\boldsymbol{y}_{\mathcal{M}} = \{y_2, y_4\}$, and $\bar{\boldsymbol{z}}_{\setminus\mathcal{M}} = \{x_1, \texttt{[M]}, x_4, \texttt{[M]}, x_6\}$. For masked n-gram $\boldsymbol{x}_{[2:4)}$, the prediction space is significantly reduced from $\mathbb{R}^{|\mathcal{V}_F|^2}$ to $\mathbb{R}^{|\langle \mathcal{V}_F, \mathcal{V}_N \rangle|}$. Explicitly n-gram MLM is performed by minimizing the negative likelihood:

$$-\log p_\theta(\boldsymbol{y}_{\mathcal{M}}|\bar{\boldsymbol{z}}_{\setminus\mathcal{M}}) = -\sum_{y \in \boldsymbol{y}_{\mathcal{M}}} \log p_\theta(y|\bar{\boldsymbol{z}}_{\setminus\mathcal{M}}). \quad (2)$$

3.3 Comprehensive N-gram Prediction

We propose to simultaneously predict n-grams in fine-grained and coarse-grained manners corresponding to single mask symbol [M], which helps to extract comprehensive n-gram semantics,

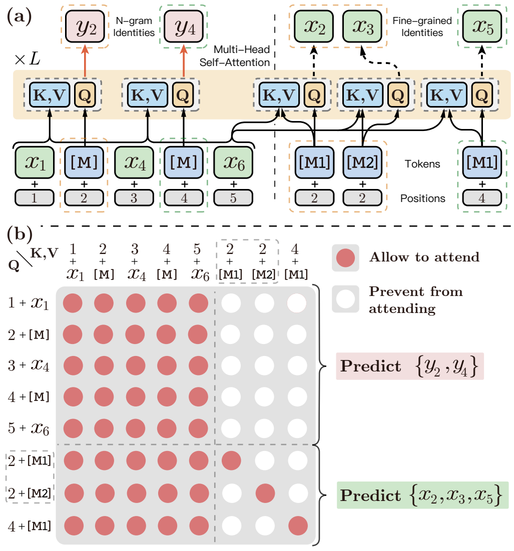

Figure 2: (a) Detailed structure of Comprehensive N-gram MLM. (b) Self-attention mask M without leaking length information of masked n-grams.

as shown in Figure 1(c). Comprehensive n-gram MLM is performed by minimizing the joint negative likelihood:

$$-\log p_\theta(\boldsymbol{y}_{\mathcal{M}}, \boldsymbol{z}_{\mathcal{M}}|\bar{\boldsymbol{z}}_{\setminus\mathcal{M}}) = -\sum_{y \in \boldsymbol{y}_{\mathcal{M}}} \log p_\theta(y|\bar{\boldsymbol{z}}_{\setminus\mathcal{M}}) - \sum_{z \in \boldsymbol{z}_{\mathcal{M}}} \sum_{x \in z} \log p_\theta(x|\bar{\boldsymbol{z}}_{\setminus\mathcal{M}}). \quad (3)$$

where the predictions of explicit n-gram $\boldsymbol{y}_{\mathcal{M}}$ and fine-grained tokens $\boldsymbol{x}_{\mathcal{M}}$ are conditioned on the same context sequence $\bar{\boldsymbol{z}}_{\setminus\mathcal{M}}$.

In detail, to predict all tokens contained in a n-gram from single [M] other than a consecutive sequence of [M], we adopt distinctive mask symbols [Mi], i $=1, ..., n$ to aggregate contextualized representations for predicting the i-th token in n-gram. As shown in Figure 2(a), along with the same position as y_2, symbols [M1] and [M2] are used as queries (Q) to aggregate representations from $\bar{\boldsymbol{z}}_{\setminus\mathcal{M}}$ (K) for the predictions of x_2 and x_3, where Q and K donate the query and key in self-attention operation (Vaswani et al., 2017). As shown in Figure 2(b), the self-attention mask metric M controls what context a token can attend to by modifying the attention weight $W_A = \texttt{softmax}(\frac{QK^T}{\sqrt{d_k}} + M)$, M is assigned as:

$$M_{ij} = \begin{cases} 0, & \text{allow to attend} \\ -\infty, & \text{prevent from attending} \end{cases} \quad (4)$$

We argue that the length information of n-grams is detrimental to the representations learning, because it will arbitrarily prune a number of semanti-

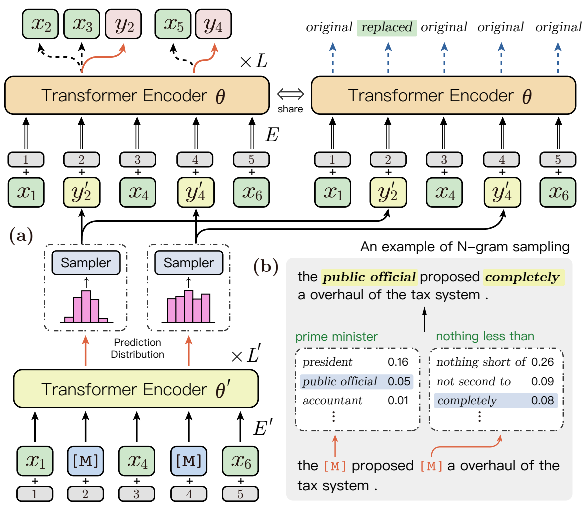

Figure 3: (a) Detailed architecture of n-gram relation modeling, where L' donates the layers of the generator model. (b) An example of plausible n-gram sampling, where dotted boxes represent the sampling module, **texts** in green are the original n-grams, and the *italic texts* in blue donate the sampled n-grams.

cally related n-grams with different lengths during predicting. From this viewpoint, for the predictions of n-gram $\{x_2, x_3\}$, 1) we prevent context $\bar{z}_{\setminus\mathcal{M}}$ from attending to $\{\texttt{[M1]}, \texttt{[M2]}\}$ and 2) prevent $\{\texttt{[M1]}, \texttt{[M2]}\}$ from attending to each other, so that the length information of n-grams will not be leaked in pre-training, as displayed in Figure 2(b).

3.4 Enhanced N-gram Relation Modeling

To explicitly learn the semantic relationships between n-grams, we jointly pre-train a small generator model θ' with explicitly n-gram MLM objective to sample plausible n-gram identities. Then we employ the generated identities to preform masking and train the standard model θ to predict the original n-grams from fake ones in coarse-grained and fine-grained manners, as shown in Figure 3(a), which is efficient to model the pair relationships between similar n-grams. The generator model θ' will not be used during fine-tuning, where the hidden size $H_{\theta'}$ of θ' has $H_{\theta'} = H_\theta/3$ empirically.

As shown in Figure 3(b), n-grams of different length can be sampled to mask original n-grams according to the prediction distributions of θ', which is more flexible and sufficient for constructing n-gram pairs than previous synonym masking methods (Cui et al., 2020) that require synonyms and original words to be of the same length. Note that our method needs a large embedding layer $E \in \mathbb{R}^{|\langle\mathcal{V}_F, \mathcal{V}_N\rangle| \times h}$ to obtain n-gram vectors in pre-training. To keep the number of parameters consistent with that of vanilla BERT, we remove the auxiliary embedding weights of n-grams during fine-tuning ($E \to E' \in \mathbb{R}^{|\mathcal{V}_F| \times h}$). Specifically, let $\boldsymbol{y}'_{\mathcal{M}}$ to be the generated n-gram identities, $\bar{z}'_{\mathcal{M}}$ to be the sequence masked by $\boldsymbol{y}'_{\mathcal{M}}$, where $\boldsymbol{y}'_{\mathcal{M}} = \{y'_2, y'_4\}$, and $\bar{\boldsymbol{z}}'_{\setminus\mathcal{M}} = \{x_1, y'_2, x_4, y'_4, x_6\}$ in Figure 3(a). The pre-training objective is to jointly minimize the negative likelihood of θ' and θ:

$$-\log p_{\theta'}(\boldsymbol{y}_{\mathcal{M}} | \bar{\boldsymbol{z}}_{\setminus\mathcal{M}}) - \log p_\theta(\boldsymbol{y}_{\mathcal{M}}, \boldsymbol{z}_{\mathcal{M}} | \bar{\boldsymbol{z}}'_{\setminus\mathcal{M}}). \quad (5)$$

Moreover, we incorporate the replaced token detection objective (RTD) to further distinguish fake n-grams from the mix-grained context $\bar{z}'_{\setminus\mathcal{M}}$ for interactions among explicit n-grams and fine-grained contextual tokens, as shown in the right part of Figure 3(a). Formally, we donate $\hat{\boldsymbol{z}}_{\setminus\mathcal{M}}$ to be the sequence after replacing masked n-grams with target n-gram identities $\boldsymbol{y}_{\mathcal{M}}$, the RTD objective is performed by minimizing the negative likelihood:

$$\begin{aligned} &- \log p_\theta\big(\mathbb{1}(\bar{\boldsymbol{z}}'_{\setminus\mathcal{M}} = \hat{\boldsymbol{z}}_{\setminus\mathcal{M}}) | \bar{\boldsymbol{z}}'_{\setminus\mathcal{M}}\big) \\ &= -\sum_{t=1}^{|\hat{\boldsymbol{z}}_{\setminus\mathcal{M}}|} \log p_\theta\big(\mathbb{1}(\bar{\boldsymbol{z}}'_{\setminus\mathcal{M},t} = \hat{\boldsymbol{z}}_{\setminus\mathcal{M},t}) | \bar{\boldsymbol{z}}'_{\setminus\mathcal{M}}, t\big). \end{aligned} \quad (6)$$

As the example depicted in Figure 3(a), the target context sequence $\hat{\boldsymbol{z}}_{\setminus\mathcal{M}} = \{x_1, y_2, x_4, y_4, x_6\}$.

3.5 N-gram Extraction

N-gram Lexicon Extraction. We employ T-test to extract semantically-complete n-grams statistically from unlabeled text corpora $\mathcal{X}$ (Xiao et al., 2020), as described in Algorithm 1. We first calculate the t-statistic scores of all n-grams appearing

---

Algorithm 1 N-gram Extraction with T-test

Input: Large-scale text corpora $\mathcal{X}$ for pre-training
Output: Semantic n-gram lexicon $\mathcal{V}_N$
▷ given initial hypothesis H_0: a randomly constructed n-gram $\boldsymbol{w} = \{x_1, ..., x_n\}$ with probability $p'(\boldsymbol{w}) = \prod_{i=1}^{n} p(x_i)$ cannot be a statistically semantic n-gram
for l in *range*(2, n) **do**
  $\mathcal{V}_{N_l} \leftarrow \langle\rangle$ ▷ initialize the lexicon for l-grams
  for l-gram $\boldsymbol{w}$ in $\mathcal{X}$ **do**
    $s \leftarrow \frac{(p(\boldsymbol{w}) - p'(\boldsymbol{w}))}{\sqrt{\sigma^2/N_l}}$: t-statistic score ▷ where statistical probability $p(\boldsymbol{w}) = \frac{\texttt{Count}(\boldsymbol{w})}{N_l}$, deviation $\sigma^2 = p(\boldsymbol{w})(1 - p(\boldsymbol{w}))$, N_l donates the count of l-grams in $\mathcal{X}$
    $\mathcal{V}_{N_l}.append(\{\boldsymbol{w}, s\})$
  $\mathcal{V}_{N_l} \leftarrow topk(\mathcal{V}_{N_l}, k_l)$ ▷ k_l is the number of l-gram
$\mathcal{V}_N \leftarrow \langle\mathcal{V}_{N_2}, ..., \mathcal{V}_{N_n}\rangle$ ▷ merge all lexicons
return $\mathcal{V}_N$

---

in $\mathcal{X}$ since the higher the t-statistic score, the more likely it is a semantically-complete n-gram. Then, we select the l-grams with the top k_l t-statistic scores to construct the final n-gram lexicon $\mathcal{V}_N$.

N-gram Boundary Extraction. To incorporate n-gram information into MLM objective, n-gram boundaries are referred to mask whole n-grams for pre-training. Given an input sequence $\boldsymbol{x} = \{x_1, ..., x_{|\boldsymbol{x}|}\}$, we employ maximum matching algorithm to traverse valid n-gram paths $\mathcal{B} = \{\boldsymbol{b}_1, ..., \boldsymbol{b}_{|\mathcal{B}|}\}$ according to $\mathcal{V}_N$, then select the shortest paths as the final n-gram boundaries $\boldsymbol{b}$, where $|\boldsymbol{b}| \leq |\boldsymbol{b}_i|, \forall i = 1, ..., |\mathcal{B}|$.

4 Experiments

In this section, we first present the pre-training configuration of ERNIE-Gram on Chinese and English text corpora. Then we compare ERNIE-Gram with previous works on various downstream tasks. We also conduct several ablation experiments to access the major components of ERNIE-Gram.

4.1 Pre-training Text Corpora

English Pre-training Data. We use two common text corpora for English pre-training:

- **Base-scale corpora:** 16GB uncompressed text from WIKIPEDIA and BOOKSCORPUS (Zhu et al., 2015), which is the original data for BERT.
- **Large-scale corpora:** 160GB uncompressed text from WIKIPEDIA, BOOKSCORPUS, OPENWEBTEXT[3], CC-NEWS (Liu et al., 2019) and STORIES (Trinh and Le, 2018), which is the original data used in RoBERTa.

Chinese Pre-training Data. We adopt the same Chinese text corpora used in ERNIE2.0 (Sun et al., 2020) to pre-train ERNIE-Gram.

4.2 Pre-training Setup

Before pre-training, we first extract 200K bi-grams and 100K tri-grams with Algorithm 1 to construct the semantic n-gram lexicon $\mathcal{V}_N$ for English and Chinese corpora. and we adopt the sub-word dictionary (30K BPE codes) used in BERT and the character dictionary used in ERNIE2.0 as our fine-grained vocabulary $\mathcal{V}_F$ in English and Chinese.

Following the previous practice, we pre-train ERNIE-Gram in base size ($L = 12, H = 768$, $A = 12$, Total Parameters=110M)[4], and set the length of the sequence in each batch up to 512 tokens. We add the relative position bias (Raffel et al., 2020) to attention weights and use Adam (Kingma and Ba, 2015) for optimizing. For pre-training on base-scale English corpora, the batch size is set to 256 sequences, the peak learning rate is $1e$-4 for 1M training steps, which are the same settings as BERT$_{\text{BASE}}$. As for large-scale English corpora, the batch size is 5112 sequences, the peak learning rate is $4e$-4 for 500K training steps. For pre-training on Chinese corpora, the batch size is 256 sequences, the peak learning rate is $1e$-4 for 3M training steps. All the pre-training hyper-parameters are supplemented in the Appendix A.

In fine-tuning, we remove the auxiliary embedding weights of explicit n-grams identities for fair comparison with previous pre-trained models.

4.3 Results on GLUE Benchmark

The General Language Understanding Evaluation (GLUE; Wang et al., 2018) is a multi-task benchmark consisting of various NLU tasks, which contains 1) pairwise classification tasks like language inference (MNLI; Williams et al., 2018, RTE; Dagan et al., 2006), question answering (QNLI; Rajpurkar et al., 2016) and paraphrase detection (QQP, MRPC; Dolan and Brockett, 2005), 2) single-sentence classification tasks like linguistic acceptability (CoLA; Warstadt et al., 2019), sentiment analysis (SST-2; Socher et al., 2013) and 3) text similarity task (STS-B; Cer et al., 2017).

The fine-tuning results on GLUE of ERNIE-Gram and various strong baselines are presented in Table 1. For fair comparison, the listed models are all in base size and fine-tuned without any data augmentation. Pre-trained with base-scale text corpora, ERNIE-Gram outperforms recent models such as TUPE and F-TFM by 1.7 and 1.3 points on average. As for large-scale text corpora, ERNIE-Gram achieves average score increase of 1.7 and 0.6 over RoBERTa and ELECTRA, demonstrating the effectiveness of ERNIE-Gram.

4.4 Results on Question Answering (SQuAD)

The Stanford Question Answering (SQuAD) tasks are designed to extract the answer span within the given passage conditioned on the question. We conduct experiments on SQuAD1.1 (Rajpurkar et al., 2016) and SQuAD2.0 (Rajpurkar et al., 2018) by adding a classification layer on the sequence outputs of ERNIE-Gram and predicting whether each token is the start or end position of the answer span.

[3] http://web.archive.org/save/http://Skylion007.github.io/OpenWebTextCorpus

[4] We donate the number of layers as L, the hidden size as H and the number of self-attention heads as A.

Models	#Param	MNLI Acc	QNLI Acc	QQP Acc	SST-2 Acc	CoLA MCC	MRPC Acc	RTE Acc	STS-B PCC	GLUE Avg
*Results of single models pre-trained on **base-scale** text corpora (16GB)*										
BERT (Devlin et al., 2019)	110M	84.5	91.7	91.3	93.2	58.9	87.3	68.6	89.5	83.1
TUPE (Ke et al., 2020)	110M	86.2	92.1	91.3	**93.3**	63.6	89.9	73.6	89.2	85.0
F-TFM$_{\text{ELECTRA}}$ (Dai et al., 2020)	110M	86.4	92.1	91.7	93.1	64.3	89.2	75.4	**90.8**	85.4
ERNIE-Gram	110M	**87.1**	**92.8**	**91.8**	93.2	**68.5**	**90.3**	**79.4**	90.4	**86.7**
*Results of single models pre-trained on **large-scale** text corpora (160GB or more)*										
XLNet (Yang et al., 2019)	110M	86.8	91.7	91.4	94.7	60.2	88.2	74.0	89.5	84.5
RoBERTa (Liu et al., 2019)	135M	87.6	92.8	91.9	94.8	63.6	90.2	78.7	91.2	86.4
ELECTRA (Clark et al., 2020)	110M	88.8	93.2	91.5	95.2	67.7	89.5	82.7	91.2	87.5
UNILMv2 (Bao et al., 2020)	110M	88.5	**93.5**	91.7	95.1	65.2	**91.8**	81.3	91.0	87.3
MPNet (Song et al., 2020)	110M	88.5	93.3	91.9	95.4	65.0	91.5	**85.2**	90.9	87.7
ERNIE-Gram	110M	**89.1**	93.2	**92.2**	**95.6**	**68.6**	90.7	83.8	**91.3**	**88.1**

Table 1: Results on the development set of the GLUE benchmark for base-size pre-trained models. Models using 16GB corpora are all pre-trained with a batch size of 256 sequences for 1M steps. STS-B and CoLA are reported by Pearson correlation coefficient (PCC) and Matthews correlation coefficient (MCC), other tasks are reported by accuracy (Acc). Note that results of ERNIE-Gram are the median of over ten runs with different random seeds.

Models	SQuAD1.1 EM	SQuAD1.1 F1	SQuAD2.0 EM	SQuAD2.0 F1
*Models pre-trained on **base-scale** text corpora (16GB)*				
BERT (Devlin et al., 2019)	80.8	88.5	73.7	76.3
RoBERTa (Liu et al., 2019)	-	90.6	-	79.7
XLNet (Yang et al., 2019)	-	-	78.2	81.0
MPNet (Song et al., 2020)	85.0	91.4	80.5	83.3
UNILMv2 (Bao et al., 2020)	85.6	92.0	80.9	83.6
ERNIE-Gram	**86.2**	**92.3**	**82.1**	**84.8**
*Models pre-trained on **large-scale** text corpora (160GB)*				
RoBERTa (Liu et al., 2019)	84.6	91.5	80.5	83.7
XLNet (Yang et al., 2019)	-	-	80.2	-
ELECTRA (Clark et al., 2020)	86.8	-	80.5	-
MPNet (Song et al., 2020)	86.8	92.5	82.8	85.6
UNILMv2 (Bao et al., 2020)	87.1	93.1	83.3	86.1
ERNIE-Gram	**87.2**	**93.2**	**84.1**	**87.1**

Table 2: Performance comparison between base-size pre-trained models on the SQuAD development sets. Exact-Match (EM) and F1 score are adopted for evaluations. Results of ERNIE-Gram are the median of over ten runs with different random seeds.

Table 2 presents the results on SQuAD for base-size pre-trained models, ERNIE-Gram achieves better performance than current strong baselines on both base-scale and large-scale pre-training text corpora.

4.5 Results on RACE and Text Classification Tasks

The ReAding Comprehension from Examinations (RACE; Lai et al., 2017) dataset collects 88K long passages from English exams at middle and high schools, the task is to select the correct choice from four given options according to the questions and passages. We also evaluate ERNIE-Gram on two large scaled text classification tasks that involve long text and reasoning, including sentiment analysis datasets IMDb (Maas et al., 2011) and topic classification dataset AG's News (Zhang et al., 2015). The results are reported in Table 3. It can be seen that ERNIE-Gram consistently outperforms previous models, showing the advantage of ERNIE-Gram on tasks involving long text and reasoning.

Models	RACE Total	RACE High	RACE Middle	IMDb Err.	AG Err.
*Pre-trained on **base-scale** text corpora (16GB)*					
BERT [a]	65.0	62.3	71.7	5.4	5.9
XLNet [b]	66.8	-	-	4.9	-
MPNet [c]	70.4	67.7	76.8	4.8	-
F-TFM$^{d}_{\text{ELECTRA}}$	-	-	-	5.2	5.4
ERNIE-Gram	**72.7**	**68.1**	75.1	**4.6**	**5.0**
*Pre-trained on **large-scale** text corpora (160GB)*					
MPNet [c]	72.0	70.3	76.3	4.4	-
ERNIE-Gram	**77.7**	**75.6**	**78.8**	**3.9**	**4.9**

Table 3: Comparison on the test sets of RACE, IMDb and AG. The listed models are all in base-size. In the results of RACE, "High" and "Middle" represent the training and evaluation sets for high schools and middle schools respectively, "Total" is the full training and evaluation set. [a] (Devlin et al., 2019); [b] (Yang et al., 2019); [c] (Song et al., 2020); [d] (Dai et al., 2020).

4.6 Results on Chinese NLU Tasks

We execute extensive experiments on six Chinese language understanding tasks, including natural language inference (XNLI; Conneau et al., 2018),

Models	XNLI Acc Dev	XNLI Acc Test	LCQMC Acc Dev	LCQMC Acc Test	DRCD EM / F1 Dev	DRCD EM / F1 Test	CMRC2018 EM / F1 Dev	DuReader EM / F1 Dev	M-NER F1 Dev	M-NER F1 Test
RoBERTa-wwn-ext$^*_{\text{LARGE}}$	82.1	81.2	90.4	87.0	89.6 / 94.8	89.6 / 94.5	68.5 / 88.4	- / -	-	-
NEZHA$_{\text{LARGE}}$ (Wei et al., 2019)	82.2	81.2	90.9	87.9	- / -	- / -	- / -	- / -	-	-
MacBERT$_{\text{LARGE}}$ (Cui et al., 2020)	82.4	81.3	90.6	87.6	91.2 / 95.6	91.7 / 95.6	70.7 / 88.9	- / -	-	-
BERT-wwn-ext$^*_{\text{BASE}}$	79.4	78.7	89.6	87.1	85.0 / 91.2	83.6 / 90.4	67.1 / 85.7	- / -	-	-
RoBERTa-wwn-ext$^*_{\text{BASE}}$	80.0	78.8	89.0	86.4	86.6 / 92.5	85.6 / 92.0	67.4 / 87.2	- / -	-	-
ZEN$_{\text{BASE}}$ (Diao et al., 2020)	80.5	79.2	90.2	88.0	- / -	- / -	- / -	- / -	-	-
NEZHA$_{\text{BASE}}$ (Wei et al., 2019)	81.4	79.3	90.0	87.4	- / -	- / -	- / -	- / -	-	-
MacBERT$_{\text{BASE}}$ (Cui et al., 2020)	80.3	79.3	89.5	87.0	89.4 / 94.3	89.5 / 93.8	68.5 / 87.9	- / -	-	-
ERNIE1.0$_{\text{BASE}}$ (Sun et al., 2019b)	79.9	78.4	89.7	87.4	84.6 / 90.9	84.0 / 90.5	65.1 / 85.1	57.9 / 72.1	95.0	93.8
ERNIE2.0$_{\text{BASE}}$ (Sun et al., 2020)	81.2	79.7	**90.9**	87.9	88.5 / 93.8	88.0 / 93.4	69.1 / 88.6	61.3 / 74.9	95.2	93.8
ERNIE-Gram$_{\text{BASE}}$	**81.8**	**81.5**	90.6	**88.5**	**90.2 / 95.0**	**89.9 / 94.6**	**74.3 / 90.5**	**64.2 / 76.8**	**96.5**	**95.3**

Table 4: Results on six Chinese NLU tasks for base-size pre-trained models. Results of models with asterisks "*" are from Cui et al., 2019. M-NER is in short for MSRA-NER dataset. "BASE" and "LARGE" donate different sizes of pre-training models. Large size models have $L = 24, H = 1024, A = 16$ and total Parameters=340M.

machine reading comprehension (CMRC2018; Cui et al., 2018, DRCD; Shao et al., 2018 and DuReader; He et al., 2018), named entity recognition (MSRA-NER; Gao et al., 2005) and semantic similarity (LCQMC; Liu et al., 2018).

Results on six Chinese tasks are presented in Table 4. It is observed that ERNIE-Gram significantly outperforms previous models across tasks by a large margin and achieves new state-of-the-art results on these Chinese NLU tasks in base-size model group. Besides, ERNIE-Gram$_{\text{BASE}}$ are also better than various large-size models on XNLI, LCQMC and CMRC2018 datasets.

4.7 Ablation Studies

We further conduct ablation experiments to analyze the major components of ERNIE-Gram.

Effect of Explicitly N-gram MLM. We compare two models pre-trained with contiguously MLM and explicitly n-gram MLM objectives in the same settings (the size of n-gram lexicon is 300K). The evaluation results for pre-training and fine-tuning are shown in Figure 4. Compared with contiguously MLM, explicitly n-gram MLM objective facilitates the learning of n-gram semantic information with lower n-gram level perplexity in pre-training and better performance on downstream tasks. This verifies the effectiveness of explicitly n-gram MLM objective for injecting n-gram semantic information into pre-training.

Size of N-gram Lexicon. To study the impact of n-gram lexicon size on model performance, we extract n-gram lexicons with size from 100K to 400K for pre-training, as shown in Figure 5. As the lexicon size enlarges, performance of contiguously MLM becomes worse, presumably because more n-grams are matched and connected as longer consecutive spans for prediction, which is more difficult for representation learning. Explicitly n-gram MLM with lexicon size being 300K achieves the best results, while the performance significantly declines when the size of lexicon increasing to 400K because more low-frequent n-grams are learning unnecessarily. See Appendix C for detailed results of different lexicon choices on GLUE and SQuAD.

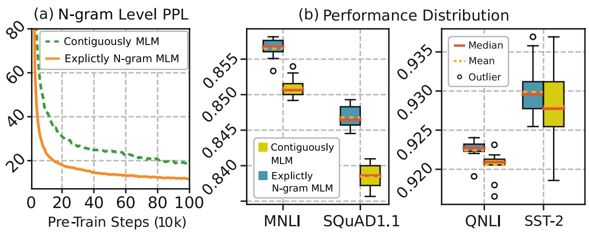

Figure 4: (a) N-gram level perplexity which is calculated by $(\prod_{i=1}^{k} \texttt{PPL}(\boldsymbol{w}_i))^{\frac{1}{k}}$ for contiguously MLM, where $\boldsymbol{w}_i$ is the i-th masked n-gram. (b) Performance distribution box plot on MNLI, QNLI, SST-2 and SQuAD1.1.

Effect of Comprehensive N-gram Prediction and Enhanced N-gram Relation Modeling. As shown in Table 5, we compare several ERNIE-Gram variants with previous strong baselines under the BERT$_{\text{BASE}}$ setting. After removing comprehensive n-gram prediction (#2), ERNIE-Gram degenerates to a variant with explicitly n-gram MLM and n-gram relation modeling and its performance drops slightly by 0.3-0.6. When removing enhanced n-gram relation modeling (#3), ERNIE-Gram degenerates to a variant with comprehen-

#	Models	MNLI		SST-2	SQuAD1.1		SQuAD2.0	
		m	mm	Acc	EM	F1	EM	F1
	XLNet [a]	85.6	85.1	93.4	-	-	78.2	81.0
	RoBERTa [b]	84.7	-	92.7	-	90.6	-	79.7
	MPNet [c]	85.6	-	**93.6**	84.0	90.3	79.5	82.2
	UNILMv2 [d]	85.6	85.5	93.0	85.0	91.5	78.9	81.8
#1	ERNIE-Gram	**86.5**	**86.4**	93.2	**85.2**	**91.7**	**80.8**	**84.0**
#2	− CNP	86.2	86.2	92.7	85.0	91.5	80.4	83.4
#3	− ENRM	85.7	85.8	93.5	84.7	91.3	79.7	82.7
#4	− CNP − ENRM	85.6	85.7	92.9	84.5	91.2	79.5	82.4

Table 5: Comparisons between comprehensive n-gram prediction (CNP) and enhanced n-gram relation modeling (ENRM) methods. All the listed models are pre-trained following the same settings of $\text{BERT}_{\text{BASE}}$ (Devlin et al., 2019) and without relative position bias. Results of ERNIE-Gram variants are the median of over ten runs with different random seeds. Results in the upper block are from [a] (Yang et al., 2019), [b] (Liu et al., 2019), [c] (Song et al., 2020) and [d] (Bao et al., 2020).

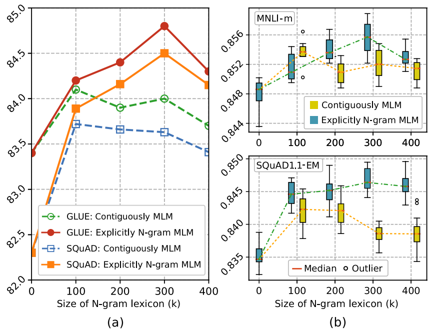

Figure 5: Quantitative study on the size of extracted n-gram lexicon. (a) Comparisons on GLUE and SQuAD. Note that SQuAD is presented by the average scores of SQuAD1.1 and SQuAD2.0. (b) Performance distribution box plot on MNLI and SQuAD1.1 datasets.

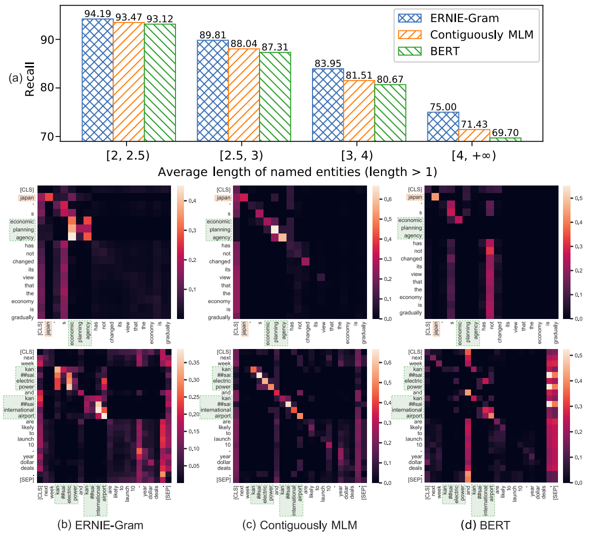

Figure 6: (a) Recall rate of whole named entities on different evaluation subsets, which have incremental average length of named entities. (b-d) Mean attention scores of 12 attention heads in the last self-attention layer. Texts in green and orange boxes are named entities standing for organizations and locations.

sive n-gram MLM and the performance drops by 0.4-1.3. If removing both comprehensive n-gram prediction and relation modeling (#4), ERNIE-Gram degenerates to a variant with explicitly n-gram MLM and the performance drops by 0.7-1.6. These results demonstrate the advantage of comprehensive n-gram prediction and n-gram relation modeling methods for efficiently n-gram semantic injecting into pre-training. The detailed results of ablation study are supplemented in Appendix C.

4.8 Case Studies

To further understand the effectiveness of our approach for learning n-grams information, we fine-tune ERNIE-Gram, contiguously MLM and lower-cased BERT on CoNLL-2003 named entity recognition task (Sang and De Meulder, 2003) for comparison. We divide the evaluation set into five subsets based on the average length of the named entities in each sentence. As shown in Figure 6(a), it is more difficult to recognize whole named entities as the length of them increases, while the performance of ERNIE-Gram declines slower than contiguously MLM and BERT, which implies that ERNIE-Gram models tighter intra-dependencies of n-grams.

As shown in Figure 6(b-d), we visualize the attention patterns in the last self-attention layer of fine-tuned models. For contiguously MLM, there are clear diagonal lines in named entities that tokens prefer to attend to themself in named entities. While for ERNIE-Gram, there are bright blocks over named entities that tokens attend to most of tokens in the same entity adequately to construct tight representation, verifying the effectiveness of ERNIE-Gram for n-gram semantic modeling.

5 Conclusion

In this paper, we present ERNIE-Gram, an explicitly n-gram masking and predicting method to eliminate the limitations of previous contiguously masking strategies and incorporate coarse-grained linguistic information into pre-training sufficiently. ERNIE-Gram conducts comprehensive n-gram prediction and relation modeling to further enhance the learning of semantic n-grams for pre-training. Experimental results on various NLU tasks demonstrate that ERNIE-Gram outperforms XLNet and RoBERTa by a large margin, and achieves state-of-the-art results on various benchmarks. Future work includes constructing more comprehensive n-gram lexicon ($n > 3$) and pre-training ERNIE-Gram with large-size model for more downstream tasks.

Acknowledgments

We would like to thank Zhen Li for his constructive suggestions, and hope everything goes well with his work. We are also indebted to the NAACL-HLT reviewers for their detailed and insightful comments on our work.

References

Hangbo Bao, Li Dong, Furu Wei, Wenhui Wang, Nan Yang, Xiaodong Liu, Yu Wang, Songhao Piao, Jianfeng Gao, Ming Zhou, et al. 2020. Unilmv2: Pseudo-masked language models for unified language model pre-training. In *Proceedings of the International Conference on Machine Learning*, pages 7006–7016.

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.

Kevin Clark, Minh-Thang Luong, Quoc V. Le, and Christopher D. Manning. 2020. Electra: Pre-training text encoders as discriminators rather than generators. In *International Conference on Learning Representations*.

Alexis Conneau, Ruty Rinott, Guillaume Lample, Adina Williams, Samuel Bowman, Holger Schwenk, and Veselin Stoyanov. 2018. XNLI: Evaluating cross-lingual sentence representations. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2475–2485, Brussels, Belgium. Association for Computational Linguistics.

Yiming Cui, Wanxiang Che, Ting Liu, Bing Qin, Shijin Wang, and Guoping Hu. 2020. Revisiting pre-trained models for Chinese natural language processing. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 657–668, Online. Association for Computational Linguistics.

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*.

Yiming Cui, Ting Liu, Li Xiao, Zhipeng Chen, Wentao Ma, Wanxiang Che, Shijin Wang, and Guoping Hu. 2018. A span-extraction dataset for chinese machine reading comprehension. *CoRR*, abs/1810.07366.

Ido Dagan, Oren Glickman, and Bernardo Magnini. 2006. The PASCAL recognising textual entailment challenge. In *Proceedings of the First International Conference on Machine Learning Challenges*, pages 177–190.

Zihang Dai, Guokun Lai, Yiming Yang, and Quoc V Le. 2020. Funnel-transformer: Filtering out sequential redundancy for efficient language processing. In *Advances in Neural Information Processing Systems*.

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.

Shizhe Diao, Jiaxin Bai, Yan Song, Tong Zhang, and Yonggang Wang. 2020. ZEN: Pre-training Chinese text encoder enhanced by n-gram representations. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 4729–4740, Online. Association for Computational Linguistics.

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

Li Dong, Nan Yang, Wenhui Wang, Furu Wei, Xiaodong Liu, Yu Wang, Jianfeng Gao, Ming Zhou, and Hsiao-Wuen Hon. 2019. Unified language model pre-training for natural language understanding and generation. In *Advances in Neural Information Processing Systems*, volume 32, pages 13063–13075. Curran Associates, Inc.

Jianfeng Gao, Mu Li, Andi Wu, and Chang-Ning Huang. 2005. Chinese word segmentation and named entity recognition: A pragmatic approach. *Computational Linguistics*, 31(4):531–574.

Wei He, Kai Liu, Jing Liu, Yajuan Lyu, Shiqi Zhao, Xinyan Xiao, Yuan Liu, Yizhong Wang, Hua Wu, Qiaoqiao She, Xuan Liu, Tian Wu, and Haifeng Wang. 2018. DuReader: a Chinese machine reading comprehension dataset from real-world applications. In *Proceedings of the Workshop on Machine Reading for Question Answering*, pages 37–46, Melbourne, Australia. 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.

Guolin Ke, Di He, and Tie-Yan Liu. 2020. Rethinking the positional encoding in language pre-training. *arXiv preprint arXiv:2006.15595*.

Diederik P. Kingma and Jimmy Ba. 2015. Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations*, San Diego, CA.

Guokun Lai, Qizhe Xie, Hanxiao Liu, Yiming Yang, and Eduard Hovy. 2017. RACE: Large-scale ReAding comprehension dataset from examinations. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 785–794, Copenhagen, Denmark. Association for Computational Linguistics.

Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. 2020. Albert: A lite bert for self-supervised learning of language representations. In *International Conference on Learning Representations*.

Xiaoya Li, Yuxian Meng, Xiaofei Sun, Qinghong Han, Arianna Yuan, and Jiwei Li. 2019. Is word segmentation necessary for deep learning of Chinese representations? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3242–3252, Florence, Italy. Association for Computational Linguistics.

Xin Liu, Qingcai Chen, Chong Deng, Huajun Zeng, Jing Chen, Dongfang Li, and Buzhou Tang. 2018. LCQMC:a large-scale Chinese question matching corpus. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 1952–1962, Santa Fe, New Mexico, USA. 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. *arXiv preprint arXiv:1907.11692*.

Andrew L. Maas, Raymond E. Daly, Peter T. Pham, Dan Huang, Andrew Y. Ng, and Christopher Potts. 2011. Learning word vectors for sentiment analysis. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 142–150, Portland, Oregon, USA. Association for Computational Linguistics.

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. *Journal of Machine Learning Research*, 21(140):1–67.

Pranav Rajpurkar, Robin Jia, and Percy Liang. 2018. Know what you don't know: Unanswerable questions for SQuAD. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 784–789, Melbourne, Australia. Association for Computational Linguistics.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. SQuAD: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2383–2392, Austin, Texas. Association for Computational Linguistics.

Erik F Sang and Fien De Meulder. 2003. Introduction to the CoNLL-2003 shared task: Language-independent named entity recognition. In *Proceedings of the Seventh Conference on Natural Language Learning at HLT-NAACL 2003*, pages 142–147.

Chih Chieh Shao, Trois Liu, Yuting Lai, Yiying Tseng, and Sam Tsai. 2018. DRCD: a chinese machine reading comprehension dataset. *arXiv preprint arXiv:1806.00920*.

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D. Manning, Andrew Ng, and Christopher Potts. 2013. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 1631–1642, Seattle, Washington, USA. Association for Computational Linguistics.

Kaitao Song, Xu Tan, Tao Qin, Jianfeng Lu, and Tie-Yan Liu. 2020. Mpnet: Masked and permuted pre-training for language understanding. In *Advances in Neural Information Processing Systems*.

Chi Sun, Xipeng Qiu, Yige Xu, and Xuanjing Huang. 2019a. How to fine-tune bert for text classification? In *China National Conference on Chinese Computational Linguistics*, pages 194–206. Springer.

Yu Sun, Shuohuan Wang, Yukun Li, Shikun Feng, Xuyi Chen, Han Zhang, Xin Tian, Danxiang Zhu, Hao Tian, and Hua Wu. 2019b. Ernie: Enhanced representation through knowledge integration. *arXiv preprint arXiv:1904.09223*.

Yu Sun, Shuohuan Wang, Yukun Li, Shikun Feng, Hao Tian, Hua Wu, and Haifeng Wang. 2020. Ernie 2.0: A continual pre-training framework for language understanding. *Proceedings of the AAAI Conference on Artificial Intelligence*, 34(05):8968–8975.

Trieu H. Trinh and Quoc V. Le. 2018. A simple method for commonsense reasoning. *ArXiv*, abs/1806.02847.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, pages 5998–6008. Curran Associates, Inc.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. 2018. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *Proceedings of the 2018 EMNLP Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 353–355, Brussels, Belgium. Association for Computational Linguistics.

Alex Warstadt, Amanpreet Singh, and Samuel R. Bowman. 2019. Neural network acceptability judgments. *Transactions of the Association for Computational Linguistics*, 7:625–641.

Junqiu Wei, Xiaozhe Ren, Xiaoguang Li, Wenyong Huang, Yi Liao, Yasheng Wang, Jiashu Lin, Xin Jiang, Xiao Chen, and Qun Liu. 2019. NEZHA: Neural contextualized representation for chinese language understanding. *arXiv preprint arXiv:1909.00204*.

Adina Williams, Nikita Nangia, and Samuel Bowman. 2018. A broad-coverage challenge corpus for sentence understanding through inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1112–1122, New Orleans, Louisiana. Association for Computational Linguistics.

Dongling Xiao, Han Zhang, Yukun Li, Yu Sun, Hao Tian, Hua Wu, and Haifeng Wang. 2020. Ernie-gen: An enhanced multi-flow pre-training and fine-tuning framework for natural language generation. In *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI-20*, pages 3997–4003. International Joint Conferences on Artificial Intelligence Organization. Main track.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R Salakhutdinov, and Quoc V Le. 2019. XLNet: Generalized autoregressive pretraining for language understanding. In *Advances in Neural Information Processing Systems*, volume 32, pages 5753–5763. Curran Associates, Inc.

Xiang Zhang, Junbo Zhao, and Yann LeCun. 2015. Character-level convolutional networks for text classification. In *Advances in Neural Information Processing Systems*, volume 28, pages 649–657. Curran Associates, Inc.

Xinsong Zhang and Hang Li. 2020. Ambert: A pre-trained language model with multi-grained tokenization. *arXiv preprint arXiv:2008.11869*.

Y. Zhu, R. Kiros, R. Zemel, R. Salakhutdinov, R. Urtasun, A. Torralba, and S. Fidler. 2015. Aligning books and movies: Towards story-like visual explanations by watching movies and reading books. In *2015 IEEE International Conference on Computer Vision (ICCV)*, pages 19–27.

A Hyperparameters for Pre-Training

As shown in Table 6, we list the detailed hyperparameters used for pre-training ERNIE-Gram on base and large scaled English text corpora and Chinese text corpora. We follow the same hyperparameters of $\text{BERT}_{\text{BASE}}$ (Devlin et al., 2019) to pre-train ERNIE-Gram on the base-scale English text corpora (16GB). We pre-train ERNIE-Gram on the large-scale text corpora (160GB) with the settings in RoBERTa (Liu et al., 2019) except the batch size being 5112 sequences.

Hyperparameters	Base-scale	Large-scale	Chinese
Layers		12	
Hidden size		768	
Attention heads		12	
Training steps	1M	500K	3M
Batch size	256	5112	256
Learning rate	1e-4	4e-4	1e-4
Warmup steps	10,000	24,000	4,000
Adam β	(0.9, 0.99)	(0.9, 0.98)	(0.9, 0.99)
Adam ϵ		1e-6	
Learning rate schedule		Linear	
Weight decay		0.01	
Dropout		0.1	
GPUs (Nvidia V100)	16	64	32

Table 6: Hyperparameters used for pre-training on different text corpora.

B Hyperparameters for Fine-Tuning

The hyperparameters for each tasks are searched on the development sets according to the average score of ten runs with different random seeds.

B.1 GLUE benchmark

The fine-tuning hyper-parameters for GLUE benchmark (Wang et al., 2018) are presented in Table 7.

Hyperparameters	GLUE
Batch size	{16, 32}
Learning rate	{5e-5, 1e-4, 1.5e-4}
Epochs	3 for MNLI and {10, 15} for others
LR schedule	Linear
Layerwise LR decay	0.8
Warmup proportion	0.1
Weight decay	0.01

Table 7: Hyperparameters used for fine-tuning on the GLUE benchmark.

B.2 SQuAD benchmark and RACE dataset

The fine-tuning hyper-parameters for SQuAD (Rajpurkar et al., 2016;Rajpurkar et al., 2018) and RACE (Lai et al., 2017) are presented in Table 8.

Hyperparameters	SQuAD	RACE
Batch size	48	32
Learning rate	{1e-4, 1.5e-4, 2e-4}	{8e-5, 1e-4}
Epochs	{2, 4}	{4, 5}
LR schedule	Linear	Linear
Layerwise LR decay	0.8	0.8
Warmup proportion	0.1	0.1
Weight decay	0.0	0.01

Table 8: Hyperparameters used for fine-tuning on the SQuAD benchmark and RACE dataset.

B.3 Text Classification tasks

Table 9 lists the fine-tuning hyper-parameters for IMDb (Maas et al., 2011) and AG'news (Zhang et al., 2015) datasets. To process texts with a length larger than 512, we follow Sun et al., 2019a to select the first 512 tokens to perform fine-tuning.

Hyperparameters	IMDb	AG'news
Batch size	32	
Learning rate	{5e-5, 1e-4, 1.5e-4}	
Epochs	3	
LR schedule	Linear	
Layerwise LR decay	0.8	
Warmup proportion	0.1	
Weight decay	0.01	

Table 9: Hyperparameters used for fine-tuning on IMDb and AG'news.

B.4 Chinese NLU tasks

The fine-tuning hyperparameters for Chinese NLU tasks including XNLI (Conneau et al., 2018), LCQMC (Liu et al., 2018), DRCD (Shao et al., 2018), DuReader (He et al., 2018), CMRC2018 and MSRA-NER (Gao et al., 2005) are presented in Table 10.

Tasks	Batch size	Learning rate	Epoch	Droput
XNLI	256	1.5e-4	3	0.1
LCQMC	32	4e-5	2	0.1
CMRC2018	64	1.5e-4	5	0.2
DuReader	64	1.5e-4	5	0.1
DRCD	64	1.5e-4	3	0.1
MSRA-NER	16	1.5e-4	10	0.1

Table 10: Hyperparameters used for fine-tuning on Chinese NLU tasks. Note that all tasks use the layerwise lr decay with decay rate 0.8.

C Detailed Results for Ablation Studies

We present the detailed results on GLUE benchmark for ablation studies in this section. The results on different MLM objectives and sizes of n-gram lexicon are presented in Table 11. The detailed

Models	Size of Lexicon	MNLI	QNLI	QQP	SST-2	CoLA	MRPC	RTE	STS-B	GLUE	SQuAD1.1		SQuAD2.0	
		Acc	Acc	Acc	Acc	MCC	Acc	Acc	PCC	Avg	EM	F1	EM	F1
$\text{BERT}_{\text{Reimplement}}$	0K	84.9	91.8	91.3	92.9	58.8	88.1	69.7	88.6	83.4	83.4	90.2	76.4	79.2
Contiguously MLM	100K	85.4	92.3	91.3	92.9	60.4	88.7	72.6	89.6	84.1	84.2	90.8	78.4	81.5
	200K	85.3	92.0	91.5	92.7	59.3	89.0	71.5	89.5	83.9	84.2	90.9	78.3	81.3
	300K	85.1	92.1	91.3	92.8	59.3	88.6	73.3	89.5	84.0	83.9	90.7	78.5	81.4
	400K	85.0	92.0	91.3	93.1	58.3	89.2	71.8	89.1	83.7	83.9	90.7	78.0	81.1
Explicitly N-gram MLM	100K	85.3	92.2	91.4	92.9	62.3	88.6	72.5	88.0	84.2	84.2	90.9	78.6	81.4
	200K	85.4	92.3	91.3	92.8	62.1	88.4	74.5	88.6	84.4	84.5	**91.3**	78.9	81.9
	300K	85.7	92.3	91.3	92.9	62.6	88.7	75.8	89.4	**84.8**	**84.7**	91.2	**79.5**	**82.4**
	400K	85.3	92.2	91.4	92.9	61.3	88.5	73.2	89.3	84.3	84.6	**91.3**	79.0	81.7

Table 11: Results on the development set of the GLUE and SQuAD benchmarks with different MLM objectives and diverse sizes of n-gram lexicon.

#	Models	MNLI		QNLI	QQP	SST-2	CoLA	MRPC	RTE	STS-B	GLUE
		m	mm	Acc	Acc	Acc	MCC	Acc	Acc	PCC	Avg
#1	$\text{ERNIE-Gram}_{\text{BASE}}$	87.1	87.1	92.8	91.8	93.2	68.5	90.3	79.4	90.4	**86.7**
#2	#1− relative position bias	86.5	86.4	92.5	91.6	93.2	68.1	90.3	79.4	90.6	86.5
#3	#2− comprehensive n-gram prediction (CNP)	86.2	86.2	92.4	91.7	92.7	65.5	90.0	78.7	90.5	86.0
#4	#2− enhanced n-gram relation modeling (ENRM)	85.7	85.8	92.6	91.2	93.5	64.8	88.9	76.9	90.0	85.5
#5	#4− comprehensive n-gram prediction (CNP)	85.6	85.7	92.3	91.3	92.9	62.6	88.7	75.8	89.4	84.8

Table 12: Comparisons between several ERNIE-Gram variants on GLUE benchmark. All the listed models are pre-trained following the same settings of $\text{BERT}_{\text{BASE}}$ (Devlin et al., 2019).

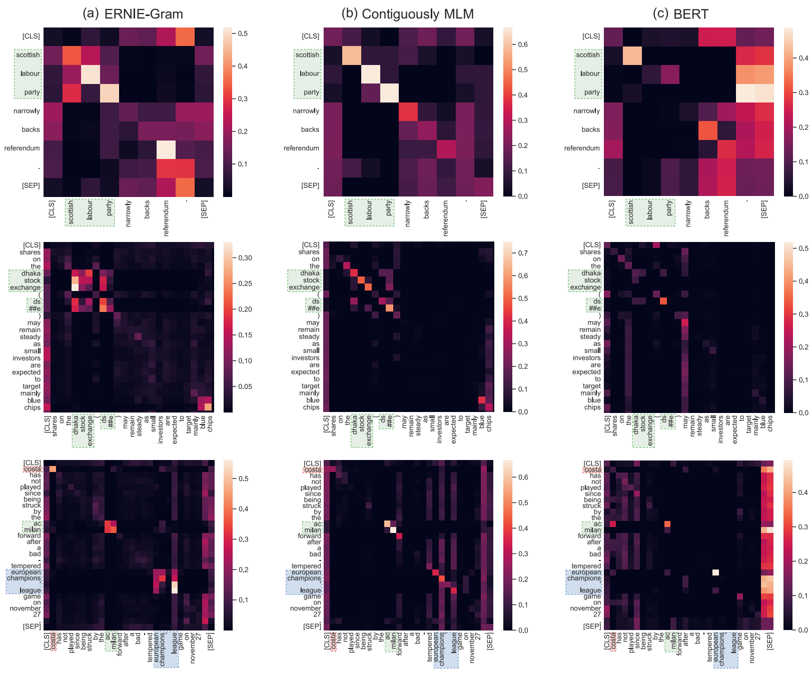

Figure 7: (a-c) Mean attention scores in the last self-attention layer. Texts in green, orange, red and blue boxes are named entities standing for organizations, locations, person and miscellaneous respectively.

results on ERNIE-Gram variants to verify the effectiveness of comprehensive n-gram prediction and enhanced n-gram relation modeling mechanisms are presented in Table 12. Results of ablation study on relative position bias (Raffel et al., 2020) are presented in Table 13.

D More cases on CoNLL2003 Dataset

We visualize the attention patterns of three supplementary cases from CoNLL2003 named entity recognition dataset (Sang and De Meulder, 2003) to compare the performance of ERNIE-Gram, contiguously MLM and BERT (lowercased), as shown

Models	MNLI		SST-2	SQuAD1.1		SQuAD2.0	
	m	mm	Acc	EM	F1	EM	F1
MPNet (Song et al., 2020)	86.2	-	**94.0**	85.0	91.4	80.5	83.3
−relative position bias	85.6	-	93.6	84.0	90.3	79.5	82.2
UNILMv2 (Bao et al., 2020)	86.1	86.1	93.2	85.6	92.0	80.9	83.6
−relative position bias	85.6	85.5	93.0	85.0	91.5	78.9	81.8
ERNIE-Gram	**87.1**	**87.1**	93.2	**86.2**	**92.3**	**82.1**	**84.8**
−relative position bias	86.5	86.4	93.2	85.2	91.7	80.8	84.0

Table 13: Ablation study on relative position bias (Raffel et al., 2020) for ERNIE-Gram and previous strong pre-trained models like MPNet and UNILMv2.

in Figure 7. For contiguously MLM, there are clear diagonal lines in named entities that tokens prefer to attend to themselves. While for ERNIE-Gram, there are bright blocks over named entities that tokens attend to most of tokens in the same entity adequately to construct tight representation.