Should You Mask 15% in Masked Language Modeling?

Alexander Wettig*, Tianyu Gao*, Zexuan Zhong, Danqi Chen
Princeton University

* equal contribution

Paper Code

Motivation

Since BERT, a **masking rate of 15%** has been adopted **ubiquitously** by its successors (RoBERTa, SpanBERT, ALBERT, DeBERTa, …)

Common belief:
- Mask > 15%: Too **little context** for model to reconstruct the sentence
- Mask < 15%: Models learns from **fewer predictions**, training is inefficient

Is masking 15% universally **optimal** for downstream performance?
What **factors** may influence the optimal masking rate?

Masking More Than 15%

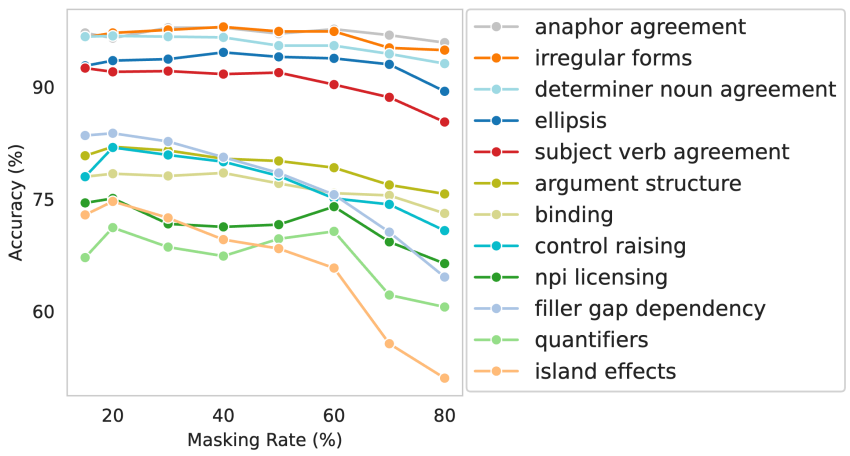
We find that 15% is **not universally optimal**

Masking 40% can outperform 15% under **an efficient pre-training recipe**

Pre-training			Fine-tuning		
m	Example	PPL	MNLI	QNLI	SQuAD[3]
15%	We study high ing rates	17.7	84.2	90.9	88.0
40%	study high ing rates	69.4	**84.5** ↑0.3	**91.6** ↑0.7	**89.8** ↑1.8
80%	high	1141.4	80.8 ↓3.4	87.9 ↓3.0	86.2 ↓1.8
	Random initialization		61.5 ↓22.7	60.9 ↓30.0	10.8 ↓77.2

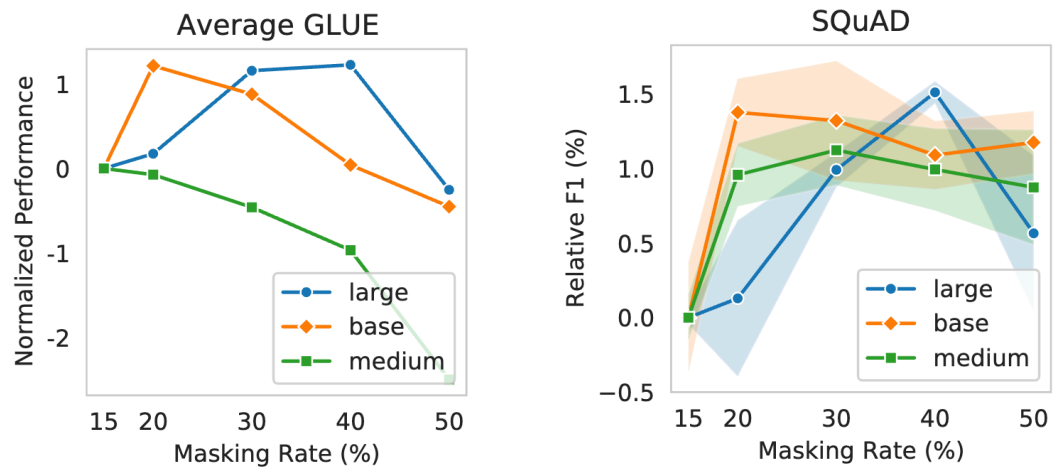
- Even if reconstructing tokens is *impossible* due to high masking rates, BERT can still learn good initialization for downstream tasks

Linguistic probing on BLIMP (via pseudo log-likelihood)



Masking Rate vs. Model Size

We pre-train and fine-tune `medium` (51M) > `base` (124M) > `large` (354M) models

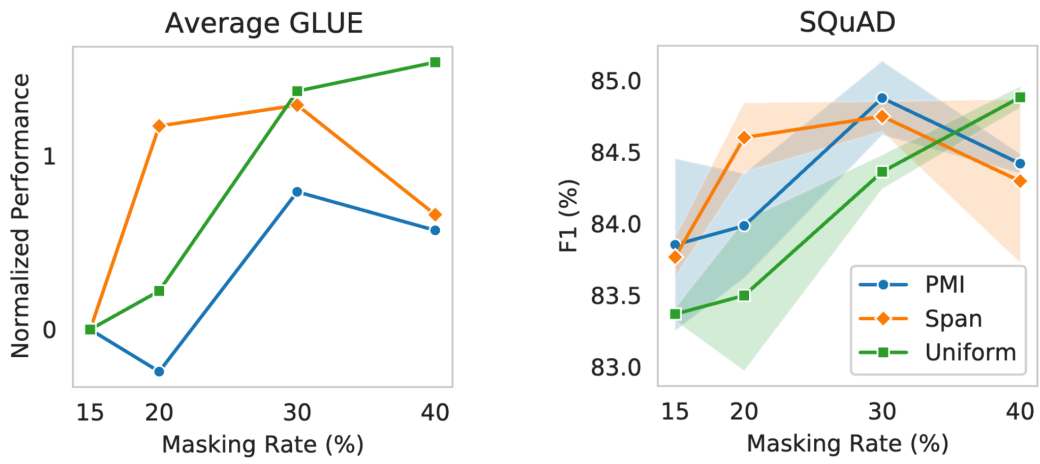


- **Larger models** can handle the **harder pre-training task** better

Masking Rate vs. Masking Strategy

Span masking[2,3] or PMI masking[4] create more challenging mask patterns to avoid local cues such as "Hong [MASK]"

These outperform uniform masking at 15% masking

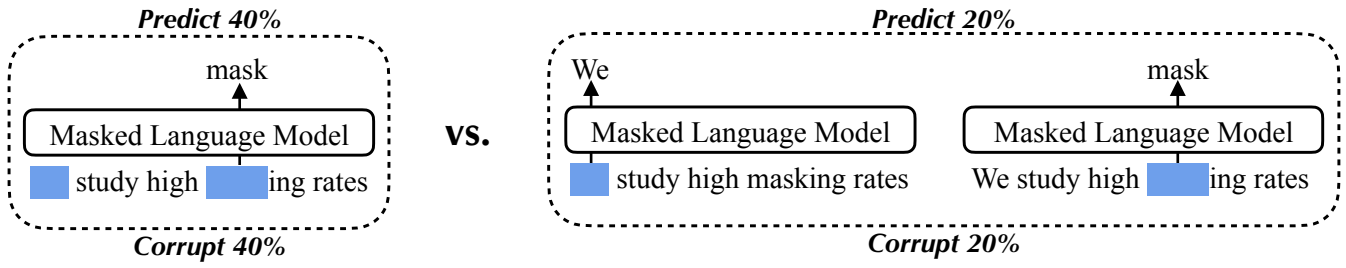


- **Uniform** masking has a **higher** optimal masking rate and can be **competitive** with **sophisticated** masking strategies

Disentangling Masking as Corruption and Prediction

The masking rate acts as both **corruption rate** and **prediction rate**.

We design **ablations** to study their individual effects:



m_{corr}	m_{pred}	MNLI	QNLI	QQP	STS-B	SST-2
40%	40%	$84.5_{0.1}$	$91.6_{0.1}$	$88.1_{0.0}$	$88.2_{0.1}$	$92.8_{0.1}$
40%	20%	83.7↓	90.6↓	87.8↓	87.5↓	92.9↑
20%	20%	84.1↓	91.3↓	87.9↓	87.4↓	92.7↓
20%	40%	85.7↑	92.0↑	87.9↓	88.6↑	93.4↑
10%	40%	86.3↑	92.3↑	88.3↑	88.9↑	93.2↑
5%	40%	86.9↑	92.2↑	88.5↑	88.6↑	93.9↑

When **controlling** for **# training sequences**:
- Higher **corruption rate** hurts the performance (slightly)
- Higher **predictions rate** helps due to more training signals and better optimization

- Benefits from more predictions can outweigh the penalty from more corruption

Revisiting 80-10-10

We also revisit **BERT's corruption strategy** of replacing:
80% of masked tokens with [MASK]
10% with a random token
10% with the original token

- Only using [MASK] performs the best on most tasks

	MNLI	QNLI	QQP	STS-B	SST-2
40% mask	$84.5_{0.1}$	$91.6_{0.1}$	$88.1_{0.0}$	$88.2_{0.1}$	$92.8_{0.1}$
+5% same	84.2↓	91.0↓	87.8↓	88.0↓	93.3↑
w/ 5% rand	84.5	91.3↓	87.9↓	87.7↓	92.6↓
w/ 80-10-10	84.3↓	91.2↓	87.9↓	87.8↓	93.0↑

References
[1] Devlin et al., *BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding*, NAACL 2019
[2] Joshi et al., *SpanBERT: Improving Pre-training by Representing and Predicting Spans*, TACL 2020
[3] Raffel et al., *Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer*, JMLR 2020
[4] Levine et al., *PMI-Masking: Principled masking of correlated spans*, ICLR 2021