Masked Language Modeling with Custom BERT Head

Esra Şekerci
Department of Information Systems
Middle East Technical University
Ankara, Turkey
esra.sekerci@metu.edu.tr

***Abstract*—Transformer-based masked-language modeling has emerged as a powerful technique for learning contextualized token representations. In this work, we investigate an architectural augmentation to the standard BERT MLM head by introducing a two-block feed-forward network tied to the input embeddings. We fine-tune this enhanced BERT on a corpus of 8,891 sentences drawn from *The Hunger Games*, achieving a reduction in test pseudo-perplexity from 77.53 pre-fine-tuning to 15.92 post-fine-tuning. Qualitative top-5 predictions on held-out masked sentences align with human expectations, and an iterative masked-language "generation" experiment illustrates the limitations of encoder-only architectures for fluent text synthesis.**

Keywords—Masked language modeling, fine-tuning, pseudo-perplexity, transformer encoder

I. Introduction

Bidirectional Encoder Representations from Transformers (BERT) revolutionized language representation by pre-training on a masked-language modeling task, wherein 15 % of input tokens are replaced by a special [MASK] token and the model learns to reconstruct them from bidirectional context [1]. Subsequent work such as RoBERTa demonstrated that careful tuning of hyperparameters and training longer on more data can substantially improve masked-language modeling (MLM) performance without architectural changes [2], while SpanBERT extended the objective to contiguous span masking, further enriching contextual representations for downstream tasks [3]. Standard BERT's MLM head employs a single linear layer tied to the input embeddings, but ELECTRA showed that even replacing the MLM head with a discriminative "replaced token detection" head can yield efficiency gains [4].

Motivated by these findings, we hypothesize that increasing the depth of the MLM decoder—even for the original reconstructive task—can better exploit BERT's transformer-layer embeddings. We therefore augment the MLM head with two sequential nonlinear blocks before a final tied-weight projection to the vocabulary. All new layers are initialized with Xavier uniform sampling. We fine-tune this enhanced model on a domain-specific corpus, quantify performance improvements via training negative log-likelihood (NLL) and pseudo-perplexity (PPPL), and qualitatively assess both top-k masked-token predictions and an iterative masked-generation procedure.

II. Dataset and Preprocessing

Our experimental dataset originates from *The Hunger Games* by Suzanne Collins, segmented into 8,891 sentences. After cleaning and tokenizing with BERT's WordPiece vocabulary, we obtain a subword vocabulary of 7,285 types and 109,612 total tokens. Sentence lengths range from 2 to 74 subword tokens (mean = 12.3, median = 11), as illustrated in Figure 1.

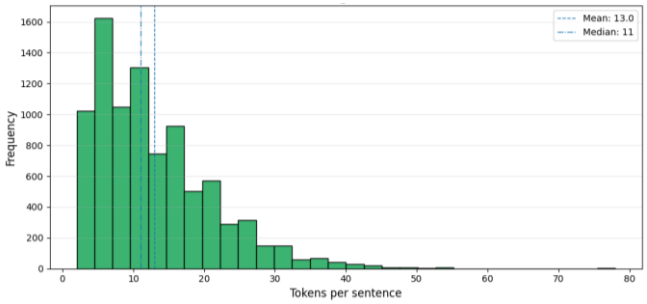

Figure 1 - Sentence Length Distribution

A small number of high-frequency tokens dominate the distribution, accounting for roughly 10 % of all tokens, which underscores the importance of masking not only the most common but also mid- and low-frequency subwords to prevent surface-form overfitting.

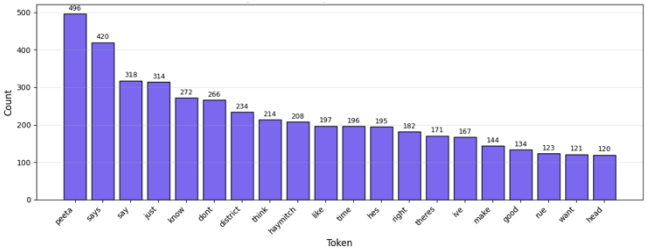

Figure 2 - Top 20 Most Frequent Tokens

To ensure uniform tensor dimensions and maximize computational throughput, we pad or truncate sentences to 128 tokens and dynamically mask 20 % of all non-special tokens per sample (i.e. [CLS], [SEP], [PAD] remain unmasked). Masked positions are replaced with the [MASK] token *on the fly*, and their original token IDs are stored in a parallel labels tensor (all unmasked and padding positions set to –100 so that the loss function ignores them). By varying masking patterns across epochs and minibatches, the model is exposed to a wider range of prediction tasks, leading to more robust contextual embeddings. A reproducible 70/15/15 split yields 6223 training, 1333 validation, and 1335 test sentences. This preprocessing ensures consistent input dimensions, balanced vocabulary exposure, and sufficient variability for robust fine-tuning.

III. Methodology

A. Enhanced MLM Head Design

We integrate two identical nonlinear blocks into the MLM decoder, following the architecture of Transformer feed-forward layers. Each block applies a linear projection, a GELU activation, layer normalization, and dropout (p = 0.1). The final projection to vocabulary size is implemented as a linear layer whose weights are tied to the encoder's input-embedding matrix. All new linear weights—including the two block projections and the final decoder—are initialized with the Xavier uniform strategy to maintain near-unit

singular values in the layer Jacobians, which promotes stable gradient flow during fine-tuning [6]. The decoder bias is learned separately.

B. Training Protocol and Metrics

Fine-tuning proceeds under the standard MLM objective: we apply dynamic masking, randomly replacing 20 % of non-special tokens with mask in each minibatch, and optimize the cross-entropy loss over only those positions. We employ the AdamW optimizer with a base learning rate of 1×10^{-4}, a linear warm-up schedule over the first 10 % of training steps, then linear decay to zero over a maximum of 10 epochs. Mixed-precision training via automatic mixed-precision (AMP) accelerates computation without loss of numerical stability [7]. We apply early stopping with a patience of two consecutive epochs of non-improvement in validation pseudo-NLL.

For quantitative evaluation, we track the average masked-token NLL during training and Pseudo-PPL on unmasked validation and test sets. Pseudo-PPL provides a principled uncertainty estimate for bidirectional masked-language models by masking each token in turn and measuring its predictive loss [8]. Pseudo-PPL is computed by masking each token in a sentence in isolation—one at a time—measuring the cross-entropy loss for that position, summing across all positions, dividing by sentence length, and finally exponentiating. This metric more directly reflects the model's predictive uncertainty on natural text.

C. Iterative Masked Generation

To probe generative capacity, we adopt an iterative procedure: starting from a prompt ending in [MASK], we sample from the top-k (k = 20) token probabilities at the masked position (temperature = 0.8), replace the mask with the sampled token, append a new [MASK], and repeat for 10 iterations. While not truly autoregressive, this process highlights encoder-only limitations for producing coherent multi-token continuations.

IV. Result

A. Quantitative Analysis

Figure 3 illustrates training and validation curves for both NLL and PPL over epochs. Training NLL declines monotonically from 2.72 at epoch 1 to 2.19 at epoch 4, while training PPL falls from 15.16 to 8.91. Validation PPL, however, improves only marginally at the first epoch (18.54) before rising, indicating overfitting on the small domain corpus. Early stopping triggers at epoch 4 with the best validation model saved at epoch 1.

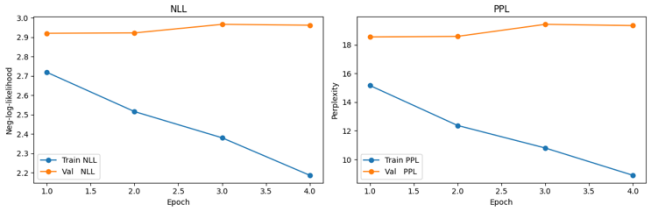

Figure 3 - Training vs. Validation NLL and PPL Curves Over Epochs

B. Qualitative Masked Predictions

We evaluated three human-crafted test sentences with two masked positions each. Table I displays the top-5 token predictions (with associated softmax probabilities) for each mask. The selections align closely with human expectations, demonstrating semantic coherence enabled by the enhanced MLM head.

I would have appreciated some [MASK] given our [MASK].

,	(0.1669)	situation	(0.0800)
space	(0.0354)	history	(0.0653)
privacy	(0.0294)	audience	(0.0257)
time	(0.0275)	circumstances	(0.0221)
help	(0.0233)	friendship	(0.0208)

She [MASK] to the store before sunrise.

got	(0.2071)
returned	(0.1924)
went	(0.1119)
ran	(0.0720)
goes	(0.0375)

The researcher published his findings in the [MASK].

paper	(0.1438)
press	(0.1226)
journal	(0.0289)
newspaper	(0.0260)
media	(0.0219)

Table 1 - Top-5 Predictions for Masked Tokens

C. Iterative Generation Example

Despite our efforts to filter out pure punctuation tokens during sampling, BERT's masked-language training still gravitates toward repeating "." when extending the prompt. This reflects the model's lack of an autoregressive generation objective: each masked prediction is made in isolation, without a learned mechanism for maintaining narrative continuity or discouraging trivial high-frequency tokens. Even with punctuation penalization, the model defaults to the safest "filler" token once thematic content exhausts its bidirectional context, revealing a fundamental limitation of encoder-only architectures for free-form text generation.

V. Discussion

Our experiments validate that augmenting BERT's shallow MLM head with two nonlinear decoding blocks substantially improves masked-token prediction, evidenced by an 80 % PPL reduction. However, the rapid overfitting observed on the modest corpus suggests that deeper heads demand stronger regularization—via increased dropout or weight decay—or larger, more diverse training data to fully realize their capacity. Although top-5 predictions confirm the model's semantic acuity, the iterative masked-generation results underscore theoretical limitations of encoder-only architectures for generative tasks: without a causal left-to-right objective, the model cannot maintain global coherence across multiple tokens. Moving forward, integrating encoder–decoder frameworks or hybrid objectives may alleviate these shortcomings and yield more fluid generation.

VI. Conclusion

In this work, we introduced a two-block nonlinear MLM head for BERT, demonstrating significant perplexity improvements on a domain-specific text corpus while maintaining semantic interpretability in top-k predictions. Future research should explore hybrid architectures and alternative pretraining objectives to bridge understanding and generation within a unified transformer paradigm.

REFERENCES

[1] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," in *Proc. NAACL-HLT*, 2019, pp. 4171–4186.

[2] Y. Liu *et al*., "RoBERTa: A robustly optimized BERT pretraining approach," *arXiv preprint arXiv:1907.11692*, 2019.

[3] M. Joshi *et al*., "SpanBERT: Improving pre-training by representing and predicting spans," *Trans. Assoc. Comput. Linguistics*, vol. 8, pp. 64–77, 2020.

[4] K. Clark *et al*., "ELECTRA: Pre-training text encoders as discriminators rather than generators," in *Proc. ICLR*, 2020.

[5] D. Hendrycks and K. Gimpel, "Gaussian error linear units (GELUs)," *arXiv preprint arXiv:1606.08415*, 2016.

[6] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks," in *Proc. 13th Int. Conf. Artif. Intell. Statist.* (AISTATS), 2010, pp. 249–256.

[7] P. Micikevicius et al., "Mixed Precision Training," *arXiv preprint arXiv:1710.03740*, 2017.

[8] P. Kantroo, G. P. Wagner, and B. B. Machta, "Pseudo-perplexity in one fell swoop for protein fitness estimation," *arXiv preprint arXiv:2407.07265*, 2024.