Masked Language Model Scoring

Julian Salazar♠ **Davis Liang**♠ **Toan Q. Nguyen**◇[*] **Katrin Kirchhoff**♠
♠ Amazon AWS AI, USA
◇ University of Notre Dame, USA
{julsal,liadavis,katrinki}@amazon.com, tnguye28@nd.edu

Abstract

Pretrained masked language models (MLMs) require finetuning for most NLP tasks. Instead, we evaluate MLMs out of the box via their *pseudo-log-likelihood scores* (PLLs), which are computed by masking tokens one by one. We show that PLLs outperform scores from autoregressive language models like GPT-2 in a variety of tasks. By rescoring ASR and NMT hypotheses, RoBERTa reduces an end-to-end LibriSpeech model's WER by 30% relative and adds up to +1.7 BLEU on state-of-the-art baselines for low-resource translation pairs, with further gains from domain adaptation. We attribute this success to PLL's unsupervised expression of linguistic acceptability without a left-to-right bias, greatly improving on scores from GPT-2 (+10 points on island effects, NPI licensing in BLiMP). One can finetune MLMs to give scores without masking, enabling computation in a single inference pass. In all, PLLs and their associated *pseudo-perplexities* (PPPLs) enable plug-and-play use of the growing number of pretrained MLMs; e.g., we use a single cross-lingual model to rerank translations in multiple languages. We release our library for language model scoring at https://github.com/awslabs/mlm-scoring.

1 Introduction

BERT (Devlin et al., 2019) and its improvements to natural language understanding have spurred a rapid succession of contextual language representations (Yang et al., 2019b; Liu et al., 2019; *inter alia*) which use larger datasets and more involved training schemes. Their success is attributed to their use of bidirectional context, often via their *masked language model* (MLM) objectives. Here, a token $\boldsymbol{w}_t$ is replaced with [MASK] and predicted using all past and future tokens $\boldsymbol{W}_{\setminus t} := (\boldsymbol{w}_1, \dots, \boldsymbol{w}_{t-1}, \boldsymbol{w}_{t+1}, \dots, \boldsymbol{w}_{|\boldsymbol{W}|})$.

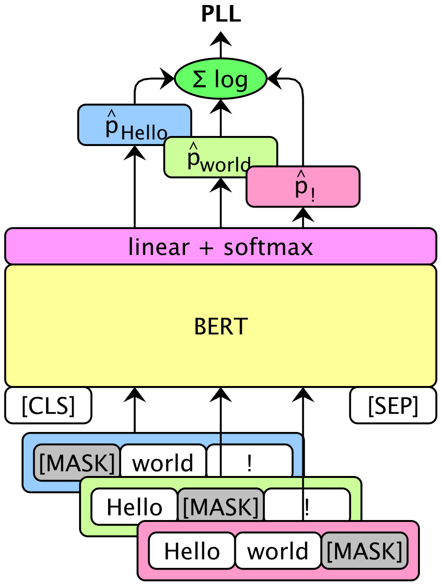

Figure 1: To score a sentence, one creates copies with each token masked out. The log probability for each missing token is summed over copies to give the pseudo-log-likelihood score (PLL). One can adapt to the target domain to improve performance, or finetune to score without masks to improve memory usage.

In contrast, conventional language models (LMs) predict $\boldsymbol{w}_t$ using only past tokens $\boldsymbol{W}_{<t} := (\boldsymbol{w}_1, \dots, \boldsymbol{w}_{t-1})$. However, this allows LMs to estimate log probabilities for a sentence $\boldsymbol{W}$ via the chain rule ($\log P_{\text{LM}}(\boldsymbol{W}) = \sum_{t=1}^{|\boldsymbol{W}|} \log P_{\text{LM}}(\boldsymbol{w}_t \mid \boldsymbol{W}_{<t})$), which can be used out of the box to rescore hypotheses in end-to-end speech recognition and machine translation (Chan et al., 2016; Gulcehre et al., 2015), and to evaluate sentences for linguistic acceptability (Lau et al., 2017).

Our work studies the corresponding *pseudo-log-likelihood scores* (PLLs) from MLMs (Wang and Cho, 2019), given by summing the conditional log probabilities $\log P_{\text{MLM}}(\boldsymbol{w}_t \mid \boldsymbol{W}_{\setminus t})$ of each sentence token (Shin et al., 2019). These are induced in BERT by replacing $\boldsymbol{w}_t$ with [MASK] (Figure 1).

[*] Work done during an internship at Amazon AWS AI.

Let Θ denote our model's parameters. Our score is

$$\mathrm{PLL}(\boldsymbol{W}) := \sum_{t=1}^{|\boldsymbol{W}|} \log P_{\mathrm{MLM}}(\boldsymbol{w}_t \mid \boldsymbol{W}_{\backslash t}; \Theta).$$

PLLs and their corresponding *pseudo-perplexities* (PPPLs) (Section 2.3) are intrinsic values one can assign to sentences and corpora, allowing us to use MLMs in applications previously restricted to conventional LM scores. Furthermore, we show that one can finetune BERT to compute PLLs in a single, non-recurrent inference pass (Section 2.2).

Existing uses of pretrained MLMs in sequence-to-sequence models for automatic speech recognition (ASR) or neural machine translation (NMT) involve integrating either their representations (Edunov et al., 2019) or weights (Yang et al., 2019a) into the encoder or decoder, then training from scratch. In contrast, we train a sequence model directly, then rescore its n-best outputs with an existing MLM. For acceptability judgments, one finetunes classifiers on MLMs using a training set (Warstadt et al., 2019; Devlin et al., 2019); PLLs enable unsupervised, relative judgements directly.

In Section 3, we show that scores from BERT compete with or even outperform GPT-2 (Radford et al., 2019), a conventional language model of similar size but trained on more data. Gains scale with dataset and model size: RoBERTa large (Liu et al., 2019) improves an end-to-end ASR model with relative WER reductions of 30%, 18% on LibriSpeech *test-clean*, *test-other* respectively (with further gains from domain adaptation), and improves state-of-the-art NMT baselines by up to +1.7 BLEU on low-resource pairs from standard TED Talks corpora. In the multilingual case, we find that the pretrained 15-language XLM (Conneau and Lample, 2019) can concurrently improve NMT systems in different target languages.

In Section 4, we analyze PLLs and propose them as a basis for other ranking/scoring schemes. Unlike log probabilities, PLL's summands are more uniform across an utterance's length (no left-to-right bias), helping disentangle fluency from likeliness. We use PLLs to perform unsupervised acceptability judgments on the BLiMP minimal pairs set (Warstadt et al., 2020); BERT and RoBERTa models improve the state-of-the-art (GPT-2 probabilities) by up to 3.9% absolute, with +10% on island effects and NPI licensing phenomena. Hence, PLLs can be used to assess linguistic competence of MLMs in a supervision-free manner.

2 Background

2.1 Pseudolikelihood estimation

Bidirectional contextual representations like BERT come at the expense of being a "true" language model $P_{\mathrm{LM}}(\boldsymbol{W})$, as there may appear no way to generate text (sampling) or produce sentence probabilities (density estimation) from these models. This handicapped their use in generative tasks, where they at best served to initialize encoder-decoder models (Edunov et al., 2019; Yang et al., 2019a) or unidirectional LMs (Wang et al., 2019).

However, BERT's MLM objective can be viewed as stochastic *maximum pseudolikelihood estimation* (MPLE) (Wang and Cho, 2019; Besag, 1975) on a training set $\mathbb{W}$, where $\{\mathbf{w}_t\}_{t=1}^{|\mathbf{W}|}$ are random variables in a fully-connected graph. This approximates conventional MLE, with MLM training asymptotically maximizing the objective:

$$\mathcal{J}_{\mathrm{PL}}(\Theta; \mathbb{W}) = \frac{1}{|\mathbb{W}|} \sum_{\boldsymbol{W} \in \mathbb{W}} \mathrm{PLL}(\boldsymbol{W}; \Theta).$$

In this way, MLMs learn an underlying joint distribution whose conditional distributions $\boldsymbol{w}_t \mid \boldsymbol{W}_{\backslash t}$ are modeled by masking at position t. We include a further discussion in Appendix B.

This enabled text generation with BERT via Gibbs sampling, leading to the proposal (but not evaluation) of a related quantity, the sum of logits, for sentence ranking (Wang and Cho, 2019). More recent work (Shin et al., 2019) sought to extend past research on future-conditional LMs in ASR (Section 5) with deeply-bidirectional self-attentive language models (bi-SANLMs). They train shallow models from scratch with the `[MASK]` scoring method, but do not relate their work to pseudolikelihood and fluency, which provide a framework to explain their success and observed behaviors.

Experimentally, we extend both works by evaluating pretrained models, domain adaptation, and usage in NMT and multilingual settings (Section 3), along with acceptability judgements and PLL's intrinsic numerical properties (Section 4).

2.2 `[MASK]`less scoring

A practical point unaddressed in both works is that computing PLLs from an MLM requires a sentence copy for each position, making the number of inference passes dependent on length (though these can be parallelized). The cost of a softmax is also incurred, which is dependent on vocabulary size

V; together this gives $\mathcal{O}(|\boldsymbol{W}| \cdot V)$. We propose reducing this to $\mathcal{O}(1)$ by training a network q with parameters Θ_S to match BERT's PLLs without `[MASK]` tokens:

$$|\mathrm{PLL}(\boldsymbol{W}) - q(\boldsymbol{W}; \Theta_S)|^2.$$

We propose finetuning q from the pretrained MLM directly (i.e., initializing Θ_S with Θ), via regression over the `[CLS]` token (Figure 2):

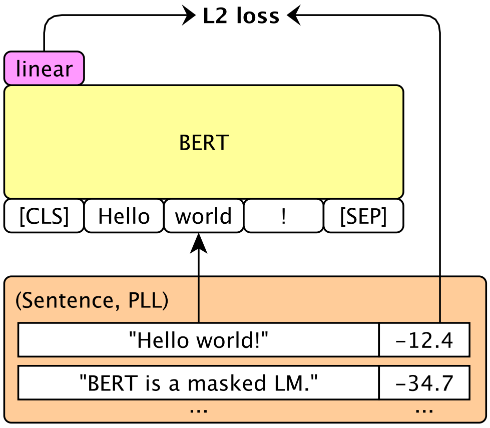

Figure 2: We learn a linear map after the `[CLS]` token against the PLLs from the pretrained MLM.

More generally, one could use any student model q as in knowledge distillation (Hinton et al., 2014). Here, the teacher gives individual token probabilities ($|\boldsymbol{W}|$ inference passes) while the student approximates their sum (one inference pass). This is reminiscent of distilling an autoregressive teacher to a parallel student, as in the case of WaveNet (Oord et al., 2018). Other `[MASK]`less bidirectional models like XLNet (Yang et al., 2019b) can also give PLL scores; we leave this to future work.

2.3 Pseudo-perplexity

Similar to conventional LMs, we also propose the *pseudo-perplexity* (PPPL) of an MLM as an intrinsic measure of how well it models a corpus of sentences $\mathbb{W}$. Let N denote the number of tokens in the corpus. Then our model's PPPL on $\mathbb{W}$ is

$$\exp\left(-\frac{1}{N}\sum_{\boldsymbol{W}\in\mathbb{W}}\sum_{t=1}^{|\boldsymbol{W}|}\log P_{\mathrm{MLM}}(\boldsymbol{w}_t \mid \boldsymbol{W}_{\setminus t})\right).$$

Past work (Chen et al., 2017) also computed this quantity with bi-RNNLMs for ASR, although such models are not deeply bidirectional like self-attentive MLMs (see Section 5).

These PPPLs can be used in lieu of perplexities. For example, during domain adaptation, one can perform early stopping with respect to development PPPL. This is in contrast to MLM accuracy, which is not a continuous loss and is often stochastic (e.g., when performing dynamic masking as in RoBERTa). In Section 4.1, we see that PPPLs naturally separate out sets of acceptable and unacceptable sentences.

Unlike previous works (Chen et al., 2017; Shin et al., 2019), we work with pre-trained BERT which is an *open-vocabulary* (subword) bidirectional LM. However, PPPLs are only comparable under the same subword vocabulary, which differs between e.g., BERT and RoBERTa. Normalizing with N as the number of *words* mitigates this. In Appendix C, we show that word-normalized PPPLs correlate with domain adaptation, and with downstream metrics like ASR and BLEU after rescoring.

3 Sequence-to-sequence rescoring

Let $\boldsymbol{X}$ denote audio features or source text tokens, and let $\boldsymbol{W} = (\boldsymbol{w}_1, \ldots, \boldsymbol{w}_{|\boldsymbol{W}|})$ denote target text tokens. For non-end-to-end ASR and MT systems, having a separate model $P_{\mathrm{LM}}(\boldsymbol{W})$ is motivated by the Bayes rule decomposition used to select the best hypothesis $\hat{\boldsymbol{W}}$ (Jelinek et al., 1975; Brown et al., 1993):

$$\begin{aligned}\hat{\boldsymbol{W}} &= \arg\max_{\mathbf{W}}[P(\mathbf{W} \mid \boldsymbol{X})] \\ &= \arg\max_{\mathbf{W}}[P(\boldsymbol{X} \mid \mathbf{W})P(\mathbf{W})].\end{aligned}$$

3.1 The log-linear model

End-to-end ASR and NMT use encoder-decoder architectures that are trained discriminatively. Though less principled, many still adopt a *log-linear model*

$$\begin{aligned}\hat{\boldsymbol{W}} &= \arg\max_{\mathbf{W}}\left[\log P(\mathbf{W} \mid \boldsymbol{X})\right] \\ &\approx \arg\max_{\mathbf{W}}\left[\log f(\mathbf{W}, \boldsymbol{X}) + \lambda \log g(\mathbf{W})\right]\end{aligned}$$

with learned functions f, g and a hyperparameter λ, to good effect (Sutskever et al., 2014; Chan et al., 2016). One often takes $f = P_{\mathrm{S2S}}(\boldsymbol{W} \mid \boldsymbol{X})$ as the sequence-to-sequence model and $g = P_{\mathrm{LM}}(\boldsymbol{W})$ as the language model. One method is *fusion*, which decomposes $f = \prod f_t$ and $g = \prod g_t$ over time (Gulcehre et al., 2015), restricting to the top N hypotheses at each time step (beam search).

In this work we consider N-best rescoring, which computes $f(\boldsymbol{W}, \boldsymbol{X})$ first, still using beam search to maintain the top N hypotheses and scores.

Then, $g(\boldsymbol{W})$ is computed for each hypothesis and interpolated with these scores, producing a new top-1 hypothesis. The sequence model is now solely responsible for "capturing" the best hypothesis $\hat{\boldsymbol{W}}$ in its beam. However, there are two advantages to N-best rescoring, which motivate PLLs as well as our maskless finetuning approach, respectively:

Decoupling of scale. Fusion requires correspondence between f_t and g_t at every t. This requires both sequence model and LM to be autoregressive and have compatible tokenizations. In rescoring, $f = P_{\text{S2S}}$ does not require g to decompose over time or to be a "true probability" at all, though g should scale with f so that λ remains valid for all lengths $|\boldsymbol{W}|$; e.g., taking $g(\boldsymbol{W})$ to be a "relevance score" between 0 and 1 would not satisfy this property. The choice of log-linear is relevant here; see Appendix B.

Length-independent inference. If g is non-recurrent, then $g(\boldsymbol{W})$ may be computed in a single inference pass. This difference manifests with self-attentive LMs like SANLMs and Transformer-XL (Dai et al., 2019), as recently explored for N-best rescoring in ASR (Li et al., 2019; Shin et al., 2019).

3.2 Experimental setup

Further implementation and experimental details can be found in Appendix A and our code release:

LMs. We rescore sequence-to-sequence hypotheses as in Section 3.1. Each hypothesis is given a log-likelihood score (uni-SANLM, GPT-2) or a PLL score (bi-SANLM, BERT, M-BERT, RoBERTa, XLM). We tune the LM weight λ on the development set to minimize word error rate (WER) for ASR or maximize tokenized BLEU for NMT. We then evaluate on the test set.

ASR. Our 100-best hypotheses are from an end-to-end, 5-layer BLSTMP model (Shin et al., 2019) from ESPnet (Watanabe et al., 2018) on the 960-hour LibriSpeech corpus (Panayotov et al., 2015). Though this baseline is not state-of-the-art, we use their lists to enable direct comparison in Table 5.

NMT. Our 100-best hypotheses are from strong Transformer baselines with BPE subwords. One was pretrained for WMT 2014 English-German (Vaswani et al., 2017); the others are state-of-the-art low-resource models we trained for five pairs from the TED Talks corpus (Qi et al., 2018) and for IWSLT 2015 English-Vietnamese (Cettolo et al., 2015), which we also describe in a dedicated, concurrent work (Nguyen and Salazar, 2019). For the low-resource models we scored tokenized hypotheses (though with HTML entities unescaped, e.g., `&quot;` ↦ `"`). Length normalization (Wu et al., 2016) is applied to NMT ($\alpha = 0.6$) and LM ($\alpha = 1.0$) scores (Section 4.3).

Corpus	**Source → target language**	**# pairs**
TED Talks	Galician (gl) → English (en)	10k
TED Talks	Slovakian (sk) → English (en)	61k
IWSLT 2015	English (en) → Vietnamese (vi)	133k
TED Talks	English (en) → German (de)	167k
TED Talks	Arabic (ar) → English (en)	214k
TED Talks	English (en) → Arabic (ar)	214k
WMT 2014	English (en) → German (de)	4.5M

Table 1: Sizes of translation datasets used in this paper.

3.3 Out-of-the-box (monolingual)

We consider BERT (Devlin et al., 2019), GPT-2 (Radford et al., 2019), and RoBERTa (Liu et al., 2019), which are trained on 17GB, 40GB, and 160GB of written text respectively. Each model comes in similarly-sized 6-layer (117M / base) and 12-layer (345M / large) versions. GPT-2 is autoregressive, while BERT and RoBERTa are MLMs. We begin by rescoring ASR outputs in Table 2.

Model	**dev**		**test**	
	clean	other	clean	other
baseline (100-best)	7.17	19.79	7.26	20.37
GPT-2 (117M, cased)	5.39	16.81	5.64	17.60
BERT (base, cased)	5.17	16.44	5.41	17.41
RoBERTa (base, cased)	**5.03**	**16.16**	**5.25**	**17.18**
GPT-2 (345M, cased)	5.15	16.48	5.30	17.26
BERT (large, cased)	4.96	16.26	5.25	16.97
RoBERTa (large, cased)	**4.75**	**15.81**	**5.05**	**16.79**
oracle (100-best)	*2.85*	*12.21*	*2.81*	*12.85*

Table 2: WERs on LibriSpeech after rescoring. Baseline lists and oracle scores are from Shin et al. (2019).

As GPT-2 is trained on cased data whereas the ASR model does not output casing or punctuation, we use cased MLMs to compare out-of-the-box performance. We see that BERT outperforms its corresponding GPT-2 despite being trained on less data. RoBERTa reduces WERs by 30% relative on LibriSpeech *test-clean* and 18% on *test-other*.

We repeat the same on English-target NMT in Table 3. As 100-best can be worse than 4-best due to the beam search curse (Yang et al., 2018; Murray and Chiang, 2018), we first decode both beam sizes to ensure no systematic degradation in our models. Hypothesis rescoring with BERT gives up to +1.1 BLEU over our strong baselines, remaining competitive with GPT-2. Using RoBERTa gives up to +1.7 BLEU over the corresponding 100-best baseline. Incidentally, we have demonstrated conclusive improvements on Transformer models via LM rescoring for the first time, despite only using N-best lists; the most recent fusion work (Stahlberg et al., 2018) only used LSTM-based models.

Model	TED Talks		
	gl→en	sk→en	ar→en
Neubig and Hu (2018)	16.2	24.0	–
Aharoni et al. (2019)	–	–	27.84
our baseline (4-best)	18.47	29.37	33.39
our baseline (100-best)	18.55	29.20	33.40
GPT-2 (117M, cased)	**19.24**	30.38	34.41
BERT (base, cased)	19.09	30.27	34.32
RoBERTa (base, cased)	19.22	**30.80**	**34.45**
GPT-2 (345M, cased)	19.16	30.76	34.62
BERT (large, cased)	19.30	30.31	34.47
RoBERTa (large, cased)	**19.36**	**30.87**	**34.73**

Table 3: Test BLEU scores on English-target language pairs from the TED Talks corpus, after rescoring.

We also consider a non-English, higher-resource target by rescoring our WMT 2014 English-German system (trained on 4.5M sentence pairs) with German BERT (base) models trained on 16GB of text (similar to English BERT)[1]. From 27.34 BLEU we get +0.5, +0.3 from uncased, cased; a diminished but present effect that can be improved as in Table 3 with more pretraining, a larger model, or domain adaptation (Section 3.5).

3.4 Out-of-the-box (multilingual)

To assess the limits of compositionality, we ask whether a shared multilingual MLM can improve translation into different target languages. We use the 100+ language M-BERT models, and the 15-language XLM models (Conneau and Lample, 2019), optionally trained with a crosslingual translation LM objective. Monolingual training was done on Wikipedia, which gives e.g., 6GB of German text; see Table 4.

The 100-language M-BERT models gave no consistent improvement. The 15-language XLMs fared better, giving +0.2-0.4 BLEU, perhaps from using language tokens and incorporating fewer languages. Our German BERT results suggest an out-of-the-box upper bound of +0.8 BLEU, as with English BERT on similar resources. We expect that increasing training data and model size will boost XLM performance, as in Section 3.3.

Model	IWSLT '15	TED Talks	
	en→vi	en→de	en→ar
Wang et al. (2018)	29.09	–	–
Aharoni et al. (2019)	–	23.31	12.95
our baseline (4-best)	31.94	30.50	13.95
our baseline (100-best)	31.84	30.44	13.94
M-BERT (base, uncased)	32.12	30.48	13.98
M-BERT (base, cased)	32.07	30.45	13.94
XLM (base*, uncased)	**32.27**	30.61	**14.13**
+ TLM objective	32.26	**30.62**	14.10
de-BERT (base, uncased)	–	**31.27**	–
de-BERT (base, cased)	–	31.22	–

Table 4: Test BLEU scores for language pairs with non-English targets, after hypothesis rescoring. base* uses 1024 hidden dims. but only 8 heads instead.

3.5 Domain adaptation

Out-of-the-box rescoring may be hindered by how closely our models match the downstream text. For example, our uncased multilingual models strip accents, exacerbating their domain mismatch with the cased, accented gold translation. We examine this effect in the setting of LibriSpeech, which has its own 4GB text corpus and is fully uncased and unpunctuated, unlike the cased MLMs in Section 3.3. We rescore using in-domain models in Table 5:

Model	dev		test	
	clean	other	clean	other
baseline (100-best)	7.17	19.79	7.26	20.37
uni-SANLM	6.08	17.32	6.11	18.13
bi-SANLM	5.52	16.61	5.65	17.44
BERT (base, Libri. only)	**4.63**	**15.56**	**4.79**	**16.50**
BERT (base, cased)	5.17	16.44	5.41	17.41
BERT (base, uncased)	5.02	16.07	5.14	16.97
+ adaptation, 380k steps	**4.37**	**15.17**	**4.58**	**15.96**
oracle (100-best)	*2.85*	*12.21*	*2.81*	*12.85*

Table 5: WERs on LibriSpeech after hypothesis rescoring. Baseline, SANLM, and oracle numbers are from Shin et al. (2019).

Using a BERT model trained only on the text corpus outperforms RoBERTa (Table 2) which is trained on far more data, underscoring the trade-off between domain matching and out-of-the-box integration. Even minor differences like casing gives +0.3-0.4 WER at test time. In Section 4.3 we see that these domain shifts can be visibly observed from the positionwise scores $\log P_{\text{MLM}}(\boldsymbol{w}_t \mid \boldsymbol{W}_{\setminus t})$.

[1] https://github.com/dbmdz/german-bert

The best results ("adaptation") still come from adapting a pretrained model to the target corpus. We proceed as in BERT, i.e., performing MLM on sequences of concatenated sentences (more details in Appendix A). In contrast, the 3-layer SANLMs (Shin et al., 2019) do per-utterance training, which is slower but may reduce mismatch even further.

Finally, we show in Appendix C that even before evaluating WER or BLEU, one can expect improvements in the downstream metric by looking at improvements in word-normalized PPPL on the target corpus. The domain-adapted MLM has lower PPPLs than the pre-trained models, and RoBERTa has lower PPPLs than BERT.

3.6 Finetuning without masking

We finetune BERT to produce scores without `[MASK]` tokens. For LibriSpeech we take the normalized text corpus and keep sentences with length $|\boldsymbol{W}| \leq 384$, score them with our adapted BERT base, then do sentence-level regression (Section 2.2). We train using Adam with a learning rate of 10^{-5} for 10 epochs (Table 6):

Model	dev	
	clean	other
baseline (100-best)	7.17	19.79
GPT-2 (117M, cased)	5.39	16.81
BERT (base, uncased, adapt.)	4.37	15.17
+ no masking	5.79	18.07
+ sentence-level finetuning	4.61	15.53

Table 6: WERs on LibriSpeech upon rescoring, showing the effects of single-copy, maskless scoring.

Sentence-level finetuning degrades performance by +0.2-0.4 WER, leaving room for future improvement. This still outperforms GPT-2 (117M, cased), though this gap might be closed by adaptation. For now, maskless finetuning could be reserved in cases where only a masked language model is available, or when latency is essential.

Remarkably, we found that out-of-the-box scoring without `[MASK]` still significantly improves the baseline. This is likely from the 20% of the time BERT does not input `[MASK]`, but instead inputs a random word or the same word (Devlin et al., 2019). Future work could explore finetuning to positionwise distributions, as in word-level knowledge distillation (Kim and Rush, 2016), for which our results are a naïve performance bound.

4 Analysis

We recall the log-linear model from Section 3.1:

$$\hat{\boldsymbol{W}} \approx \arg\max_{\mathbf{W}} [\log f(\mathbf{W}, \boldsymbol{X}) + \lambda \log g(\mathbf{W})]$$

Although end-to-end models $f = P_{\text{S2S}}(\boldsymbol{W}|\boldsymbol{X})$ predict $\boldsymbol{W}$ directly from $\boldsymbol{X}$, interpolation with the unconditional $g(\boldsymbol{W})$ remains helpful (Toshniwal et al., 2018). One explanation comes from *cold* and *simple fusion* (Sriram et al., 2018; Stahlberg et al., 2018), which further improve on shallow fusion (Section 3.1) by learning $g(\boldsymbol{W})$ first. They argue g expresses *fluency*; fixing g early allows $f(\boldsymbol{W}, \boldsymbol{X})$ to focus its capacity on *adequacy* in encoding the source, disentangling the two. With this perspective in mind, we compare $\log P_{\text{LM}}$ and PLL as candidates for g.

4.1 Relative linguistic acceptability

In this work we interpret fluency as linguistic *acceptability* (Chomsky, 1957); informally, the syntactic and semantic validity of a sentence according to human judgments (Schütze, 1996). Its graded form is well-proxied by neural language model scores $\log P_{\text{LM}}$ once length and lexical content are controlled for (Lau et al., 2017). This can be seen in a controlled setting using *minimal pairs* and GPT-2 (345M) scores:

✓	*Raymond is selling this sketch.*	-40.0
	Raymond is selling this sketches.	-45.2

This example is from the Benchmark of Linguistic Minimal Pairs (BLiMP) (Warstadt et al., 2020), a challenge set of 67k pairs which isolate contrasts in syntax, morphology, and semantics (in this case, determiner-noun agreement). While its predecessor, the Corpus of Linguistic Acceptability (CoLA), has a training set and asks sentences to be labeled "acceptable" or not in isolation (Warstadt et al., 2019), BLiMP provides an unsupervised setting: language models are evaluated on how often they give the acceptable sentence a higher (i.e., less negative) score. This is equivalent to 2-best rescoring without sequence model scores ($f = 0$). Since most minimal pairs only differ by a single word, the effect of length on log probabilities and PLL (which we discuss in Section 4.3) is mitigated.

We compute PLLs on the sentences of each pair using BERT and RoBERTa models, and choose the sentence with the highest score. Our results are in Table 7. Despite using less than half the data and

Model (cased)	**Overall**	ANA. AGR	ARG. STR	BINDING	CTRL. RAIS.	D-N AGR	ELLIPSIS	FILLER GAP	IRREGULAR	ISLAND	NPI	QUANTIFIERS	S-V AGR	Unacc. PPPL	Acc. PPPL	Ratio
GPT-2 (345M)	82.6	**99.4**	83.4	77.8	83.0	96.3	86.3	81.3	94.9	71.7	74.7	**74.1**	88.3	–	–	–
BERT (base)	84.2*	97.0	80.0	**82.3***	79.6	**97.6***	89.4*	83.1*	96.5*	73.6*	84.7*	71.2	**92.4***	111.2	59.2	1.88
BERT (large)	84.8*	97.2	80.7	82.0*	82.7	**97.6***	86.4	84.3*	92.8	77.0*	83.4*	72.8	91.9*	128.1	63.6	2.02
RoBERTa (base)	85.4*	97.3	83.5	77.8	81.9	97.0	**91.4***	**90.1***	96.2*	80.7*	81.0*	69.8	91.9*	213.5	87.9	2.42
RoBERTa (large)	**86.5***	97.8	**84.6***	79.1*	**84.1***	96.8	90.8*	88.9*	**96.8***	**83.4***	**85.5***	70.2	91.4*	194.0	77.9	2.49
Human	*88.6*	*97.5*	*90.0*	*87.3*	*83.9*	*92.2*	*85.0*	*86.9*	*97.0*	*84.9*	*88.1*	*86.6*	*90.9*	–	–	–

Table 7: Unsupervised performance (forced choice accuracy) on BLiMP using log probabilities (GPT-2) or PLLs. Human scores from Warstadt et al. (2020). Values with * denote improvements over GPT-2 of $\geq$1% absolute.

a third of the capacity, BERT (base) already outperforms the previous state of the art (GPT-2) by 1.6% absolute, increasing to 3.9% with RoBERTa (large). There are 4 of 12 categories where all four PLLs outperform the log probabilities by $\geq$1% absolute (values marked by *), and 7 where three or more PLLs outperform by this margin. Interestingly, PLLs do consistently worse on quantifiers, though all are relatively bad versus the human baseline. The ratio of token-level PPPLs between unacceptable and acceptable sentences increases with performance, separating the two sentence sets.

RoBERTa improves by around 10% on filler-gap dependencies, island effects, and negative polarity items (NPIs), largely closing the human gap. This suggests that the difficulty of these BLiMP categories was due to P_{LM} decomposing autoregressively, and not intrinsic to unsupervised language model training, as the original results may suggest (Warstadt et al., 2020). For some intuition, we include examples in Table 8. In the subject-verb agreement example, BERT sees *The pamphlets* and *resembled those photographs* when scoring *have* vs. *has*, whereas GPT-2 only sees *The pamphlets*, which may not be enough to counter the misleading adjacent entity *Winston Churchill* at scoring time.

4.2 Interpolation with direct models

We observe that $g = \text{PLL}(\boldsymbol{W})$ is not unduly affected by unconditional token frequencies, which mitigates degradation in adequacy at interpolation time. Specifically, consider a two-word proper noun (e.g., $\boldsymbol{W}$ = "San Francisco"):

$$\begin{aligned}
&\log P_{\text{LM}}(\boldsymbol{W}) \\
&= \log P_{\text{LM}}(\text{San}) + P_{\text{LM}}(\text{Francisco} \mid \text{San}) \\
&\ll \log P_{\text{MLM}}(\text{San} \mid \text{Francisco}) \\
&\qquad + \log P_{\text{MLM}}(\text{Francisco} \mid \text{San}) \\
&= \text{PLL}(\boldsymbol{W}).
\end{aligned}$$

It is a highly-fluent but low-probability bigram and thus gets penalized by $\log P_{\text{LM}}(\boldsymbol{W})$. Informally, $\text{PLL}(\boldsymbol{W})$ expresses how likely each token is given other tokens (self-consistency), while $\log P_{\text{LM}}(\boldsymbol{W})$ expresses the unconditional probability of a sentence, beginning with the costly unconditional term $P_{\text{LM}}(\text{San})$. We see this in practice when we take LM to be GPT-2 (345M) and MLM to be RoBERTa (large). Substituting in the actual scores:

$$\begin{aligned}
\log P_{\text{GPT-2}}(\boldsymbol{W}) &= -8.693 \\
&= (-7.749) + (-0.944) \\
&\ll (-0.006) + (-1.000) \\
&= -1.006 = \text{PLL}_{\text{RoBERTa}}(\boldsymbol{W}).
\end{aligned}$$

Both give similar probabilities $P(\text{Francisco} \mid \text{San}) \approx e^{-1.0} \approx 37\%$, but differ in the first summand.

We examine the interplay of this bias with our sequence models, in cases where the baseline, GPT-2, and BERT gave different top-1 hypotheses (Table 8). In our examples, GPT-2 restores fluency using common and repeated words, at the cost of adequacy:

clasping truth and ↦ *class in truth and*,
Union by the Union Sivities ↦
Union by the Union by the Union Civities.

One can view these as exacerbations of the rare-word problem due to overconfident logits (Nguyen and Chiang, 2018), and of over-translation (Tu et al., 2016). Meanwhile, BERT rewards self-consistency, which lets rarer but still-fluent words with better acoustic or translation scores to persist:

clasping truth and ↦ *clasping truth in*,
Union by the Union Sivities ↦
Union by the Union of LiberCivities,

System	Model	Output sentence
BLiMP (S-V agreement)	BERT	The pamphlets about Winston Churchill **have** resembled those photographs.
	GPT-2	The pamphlets about Winston Churchill **has** resembled those photographs.
BLiMP (islands)	BERT	Who does Amanda find **while thinking about** Lucille?
	GPT-2	Who does Amanda find Lucille **while thinking about**?
LibriSpeech (dev-other)	Baseline	**clasping truth and jail ya in** the mouth of the student is that building up or tearing down
	GPT-2	**class in truth and jail ya in** the mouth of the student is that building up or tearing down
	BERT (adapted)	**clasping truth in jail gagging** the mouth of the student is that building up or tearing down
	Target	**clapping truth into jail gagging** the mouth of the student is that building up or tearing down
gl→en (test)	Source (gl)	Traballaba de asesora científica na **ACLU , a Unión polas Liberdades Civís** .
	Baseline	I worked on a scientific status on the **ACL, the Union by the Union Sivities** .
	GPT-2	I worked on a scientific status on the **ACL, the Union by the Union by the Union Civities** .
	BERT	I worked on a scientific status on the **ACL, the Union by the Union of LiberCivities** .
	Target (en)	I was working at the **ACLU** as the organization 's science advisor .

Table 8: Examples of different top-1 hypotheses after ranking the minimal pairs / rescoring hypotheses from 4-best models, with differences highlighted. GPT-2 and BERT both promote fluency, but GPT-2's unconditional $P_{\text{LM}}(\boldsymbol{W})$ scores cause it to overweight common word sequences at the expense of adequacy.

which preserves the *p* sound in the ground truth (*clapping*) for ASR, and promotes the more globally-fluent *Union by the Union **of** LiberCivities*. We also the under-translation (i.e., omission) of *Liber* being corrected, without being discouraged by the rare sequence *LiberCivities*.

Given the differences between PLLs and log probabilities, we explore whether ensembling both improves performance in Appendix D. Similar to the largely-dominant results of MLMs on BLiMP over GPT-2 (Section 4.1), we find that as the MLM gets stronger, adding GPT-2 scores has negligible effect, suggesting their roles overlap.

4.3 Numerical properties of PLL

PLL's numerical properties make it an ideal foundation for future ranking or scoring schemes. For example, given fixed $|\boldsymbol{W}|$ one expects $-\log P_{\text{MLM}}(\boldsymbol{w}_t \mid \boldsymbol{W}_{\setminus t})$ to be in the same range for all t. Meanwhile $-\log P_{\text{LM}}(\boldsymbol{w}_t \mid \boldsymbol{W}_{<t})$ decreases as $t \to |\boldsymbol{W}|$, the rate of which was studied in recurrent language models (Takahashi and Tanaka-Ishii, 2018). We validate this with GPT-2 (Figure 3) and BERT (Figure 4). In particular, we see the outsized cost of the unconditional first unigram in Figure 3. These also explain observations that bi-SANLM was more robust than uni-SANLM at shorter and earlier positions (Shin et al., 2019); the difference is intrinsic to log probabilities versus PLLs, and is not due to model or data size.

Figure 4 also shows domain adaptation (Section 3.5) affects PLL's positionwise cross-entropies. Cased BERT spikes at position 1, as it observes a lowercased word where a capitalized word is expected. All models spike at the final word of an utterance before the period. Periods are difficult to predict in English in general, but here more so due to a mismatch between the BERT+LibriSpeech text corpora and the LibriSpeech test set, as the latter is segmented by voice activity and not punctuation (Panayotov et al., 2015). Otherwise the cross-entropies are largely flat. This, along with

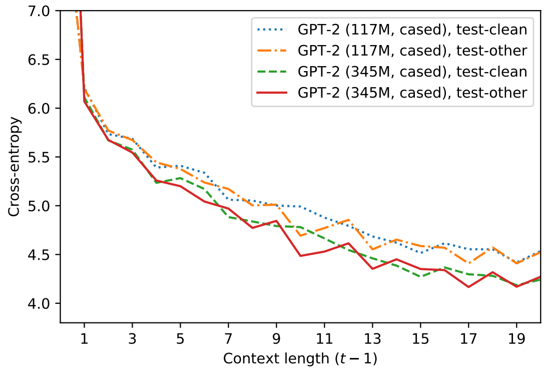

Figure 3: Cross-entropy (natural base) of $\boldsymbol{w}_t \mid \boldsymbol{W}_{<t}$ versus context length $(t-1)$ from GPT-2 models, averaged over LibriSpeech's test utterances.

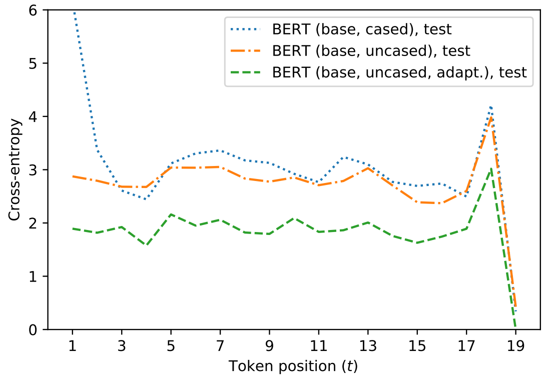

Figure 4: Cross-entropy (natural base) of $\boldsymbol{w}_t \mid \boldsymbol{W}_{\setminus t}$ versus t from BERT, averaged over LibriSpeech's 189 test utterances of length $|\boldsymbol{W}| = 19$ (including ".").

our success on BLiMP, suggests positionwise probabilities as a way of detecting "disfluencies" (at least, those in the form of domain mismatch) by observing spikes in cross-entropy; with $\log P_{\text{LM}}$, spikes are confounded by the curve in Figure 3.

In Appendix C we plot sentence-level PLLs versus $|\boldsymbol{W}|$ and observe linearity as $|\boldsymbol{W}| \to \infty$, with spikes from the last word and uncapitalized first word averaging out. This behavior motivates our choice of $\alpha = 1.0$ for the Google NMT-style length penalty (Wu et al., 2016) for PLLs, which corresponds to the asymptotically-linear $\text{LP}_{\text{MLM}} = (5 + |\boldsymbol{W}|)/(5 + 1)$. In contrast, autoregressive scores like $P_{\text{LM}}(\boldsymbol{W})$ integrate over the inverse power-law curve in Figure 3. We speculate that this explains the effectiveness of their hyperparameter $\alpha = 0.6$, widely used in NMT baselines like ours, as there exists C such that

$$\text{LP}_{\text{S2S}}(\boldsymbol{W}) = \frac{(5 + |\boldsymbol{W}|)^{0.6}}{(5+1)^{0.6}} \approx \int_0^{|\boldsymbol{W}|} \frac{C}{(5+x)^{0.4}}\, \mathrm{d}x.$$

5 Related work

Our work extends the closest previous works (Wang and Cho, 2019; Shin et al., 2019) with regards to experiments and tasks, as outlined in Section 2.1. Furthermore, neither work considers the inference cost of masked rescoring, which we address with our maskless scoring approach, or analyze PLL's numerical properties.

Future context. Log probabilities conditioned on past and future context have been used in MT (Finch and Sumita, 2009; Xiong et al., 2011) and perennially in ASR (Shi et al., 2013; Arisoy et al., 2015; Chen et al., 2017) to positive effect. However, these are not "deep bidirectional" in that they model interactions between $\boldsymbol{W}_{<t}$ and $\boldsymbol{W}_{>t}$ via the forward and backward context vectors, whereas MLMs model all pairwise interactions $\boldsymbol{w}_s$ and $\boldsymbol{w}_{s'}$ via dot-product attention (c.f. ELMo versus BERT). Their PLLs would have different properties from ours (e.g., their cross-entropies in Figure 4 would be convex, not flat).

Discriminative language modeling. Previous works (Roark et al., 2004; Huang et al., 2018) have explored training language models that directly optimize for a downstream metric (WER, BLEU). While we also eschew using log probabilities from conventional LMs, our approach remains generative. Log probabilities model the joint distribution; PLL does so as well, albeit implicitly (Appendix B). Its summands (conditional probabilities) remain accessible for Gibbs sampling, and it is not tailored to any metric. The two approaches are complementary; for example, one could use PLLs as a "prior" or regularizer for scores given by discriminatively-finetuned BERT models in tasks like passage re-ranking (Nogueira and Cho, 2019).

Language model integration. Beyond fusion and initialization from pretrained MLMs, monolingual pretraining has improved NMT performance (Ramachandran et al., 2017; Conneau and Lample, 2019). However, compositional integrations of language representation models remain prevalent, especially in ASR. Contemporary examples are the use of finetuned BERT scores in a question-answering pipeline (Nogueira and Cho, 2019) or "as-is" cosine similarity scores from BERT to evaluate generated text (Zhang et al., 2020). For example, in decoder pretraining one might have no pretrained multilingual LMs, which are difficult to train (Ragni et al., 2016). However, one may have an M-BERT or XLM adapted to a target language or domain. Finally, note that N-best rescoring is not mutually exclusive with pretraining, although pretraining may already go some of the way in improving fluency.

6 Conclusion

We studied masked language model scores, or pseudo-log-likelihoods, in a variety of seetings. We showed the effectiveness of N-best rescoring with PLLs from pretrained MLM in modern sequence-to-sequence models, for both ASR and low- to medium-resource NMT. We found rescoring with PLLs can match or outperform comparable scores from large unidirectional language models (GPT-2). We attributed this to PLL's promotion of fluency via self-consistency, as demonstrated by improvements on unsupervised accessibility judgement tasks and qualitative analysis. We examined the numerical properties of PLLs, proposed maskless scoring for speed, and proposed PPPLs for intrinsic evaluation of MLMs, releasing a codebase demonstrating our work. Future work could find additional compositional uses of MLMs, simplify non-masked PLL computations, and use PLLs to devise better sentence- or document-level scoring metrics.

Acknowledgments

We thank Phillip Keung and Chris Varano for their thoughtful suggestions on this work.

References

Roee Aharoni, Melvin Johnson, and Orhan Firat. 2019. Massively multilingual neural machine translation. In *NAACL-HLT*.

Ebru Arisoy, Abhinav Sethy, Bhuvana Ramabhadran, and Stanley Chen. 2015. Bidirectional recurrent neural network language models for automatic speech recognition. In *ICASSP*.

Julian Besag. 1975. Statistical analysis of non-lattice data. *The Statistician*, 24(3):179–195.

Peter F Brown, Vincent J Della Pietra, Stephen A Della Pietra, and Robert L Mercer. 1993. The mathematics of statistical machine translation: Parameter estimation. *Computational Linguistics*, 19(2):263–311.

Mauro Cettolo, Jan Niehues, Sebastian Stüker, Luisa Bentivogli, R Cattoni, and Marcello Federico. 2015. The IWSLT 2015 evaluation campaign. Technical report, FBK and KIT.

William Chan, Navdeep Jaitly, Quoc Le, and Oriol Vinyals. 2016. Listen, attend and spell: A neural network for large vocabulary conversational speech recognition. In *ICASSP*.

Xie Chen, Anton Ragni, Xunying Liu, and Mark JF Gales. 2017. Investigating bidirectional recurrent neural network language models for speech recognition. In *INTERSPEECH*.

Noam Chomsky. 1957. *Syntactic structures*. Mouton.

Alexis Conneau and Guillaume Lample. 2019. Cross-lingual language model pretraining. In *NeurIPS*.

Zihang Dai, Zhilin Yang, Yiming Yang, Jaime Carbonell, Quoc V Le, and Ruslan Salakhutdinov. 2019. Transformer-XL: Attentive language models beyond a fixed-length context. In *ACL*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *NAACL-HLT*.

Sergey Edunov, Alexei Baevski, and Michael Auli. 2019. Pre-trained language model representations for language generation. In *NAACL-HLT*.

Andrew Finch and Eiichiro Sumita. 2009. Bidirectional phrase-based statistical machine translation. In *EMNLP*.

Caglar Gulcehre, Orhan Firat, Kelvin Xu, Kyunghyun Cho, Loic Barrault, Huei-Chi Lin, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2015. On using monolingual corpora in neural machine translation. *arXiv preprint arXiv:1503.03535*.

Jian Guo, He He, Tong He, Leonard Lausen, Mu Li, Haibin Lin, Xingjian Shi, Chenguang Wang, Junyuan Xie, Sheng Zha, et al. 2020. GluonCV and GluonNLP: Deep learning in computer vision and natural language processing. *JMLR*, 21(23):1–7.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. 2014. Distilling the knowledge in a neural network. *Deep Learning Workshop, NeurIPS*.

Jiaji Huang, Yi Li, Wei Ping, and Liang Huang. 2018. Large margin neural language model. In *EMNLP*.

Frederick Jelinek, Lalit Bahl, and Robert Mercer. 1975. Design of a linguistic statistical decoder for the recognition of continuous speech. *IEEE Trans. Inf. Theory*, 21(3):250–256.

Yoon Kim and Alexander M Rush. 2016. Sequence-level knowledge distillation. In *EMNLP*.

Jey Han Lau, Alexander Clark, and Shalom Lappin. 2017. Grammaticality, acceptability, and probability: A probabilistic view of linguistic knowledge. *Cognitive Science*, 41(5):1202–1241.

Jason Li, Vitaly Lavrukhin, Boris Ginsburg, Ryan Leary, Oleksii Kuchaiev, Jonathan M Cohen, Huyen Nguyen, and Ravi Teja Gadde. 2019. Jasper: An end-to-end convolutional neural acoustic model. In *INTERSPEECH*.

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*.

Kenton Murray and David Chiang. 2018. Correcting length bias in neural machine translation. In *WMT*.

Graham Neubig and Junjie Hu. 2018. Rapid adaptation of neural machine translation to new languages. In *EMNLP*.

Toan Q Nguyen and David Chiang. 2018. Improving lexical choice in neural machine translation. In *NAACL-HLT*.

Toan Q Nguyen and Julian Salazar. 2019. Transformers without tears: Improving the normalization of self-attention. In *IWSLT*.

Rodrigo Nogueira and Kyunghyun Cho. 2019. Passage re-ranking with BERT. *arXiv preprint arXiv:1901.04085*.

Aaron van den Oord, Yazhe Li, Igor Babuschkin, Karen Simonyan, Oriol Vinyals, Koray Kavukcuoglu, George van den Driessche, Edward Lockhart, Luis C Cobo, Florian Stimberg, et al. 2018. Parallel WaveNet: Fast high-fidelity speech synthesis. In *ICML*.

Vassil Panayotov, Guoguo Chen, Daniel Povey, and Sanjeev Khudanpur. 2015. LibriSpeech: An ASR corpus based on public domain audio books. In *ICASSP*.

Ye Qi, Devendra Singh Sachan, Matthieu Felix, Sarguna Janani Padmanabhan, and Graham Neubig. 2018. When and why are pre-trained word embeddings useful for neural machine translation? In *NAACL-HLT*.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners. Technical report, OpenAI.

Anton Ragni, Edgar Dakin, Xie Chen, Mark JF Gales, and Kate M Knill. 2016. Multi-language neural network language models. In *INTERSPEECH*.

Prajit Ramachandran, Peter J Liu, and Quoc V Le. 2017. Unsupervised pretraining for sequence to sequence learning. In *EMNLP*.

Brian Roark, Murat Saraclar, Michael Collins, and Mark Johnson. 2004. Discriminative language modeling with conditional random fields and the perceptron algorithm. In *ACL*.

Carson T Schütze. 1996. *The empirical base of linguistics: Grammaticality judgments and linguistic methodology*. Language Science Press.

Yangyang Shi, Martha Larson, Pascal Wiggers, and Catholijn M Jonker. 2013. Exploiting the succeeding words in recurrent neural network language models. In *INTERSPEECH*.

Joongbo Shin, Yoonhyung Lee, and Kyomin Jung. 2019. Effective sentence scoring method using BERT for speech recognition. In *ACML*.

Anuroop Sriram, Heewoo Jun, Sanjeev Satheesh, and Adam Coates. 2018. Cold fusion: Training seq2seq models together with language models. In *INTERSPEECH*.

Felix Stahlberg, James Cross, and Veselin Stoyanov. 2018. Simple fusion: Return of the language model. In *WMT*.

Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. Sequence to sequence learning with neural networks. In *NeurIPS*.

Shuntaro Takahashi and Kumiko Tanaka-Ishii. 2018. Cross entropy of neural language models at infinity–a new bound of the entropy rate. *Entropy*, 20(11):839.

Shubham Toshniwal, Anjuli Kannan, Chung-Cheng Chiu, Yonghui Wu, Tara N Sainath, and Karen Livescu. 2018. A comparison of techniques for language model integration in encoder-decoder speech recognition. In *SLT*.

Zhaopeng Tu, Zhengdong Lu, Yang Liu, Xiaohua Liu, and Hang Li. 2016. Modeling coverage for neural machine translation. In *ACL*.

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 *NeurIPS*.

Alex Wang and Kyunghyun Cho. 2019. BERT has a mouth, and it must speak: BERT as a Markov random field language model. In *NeuralGen*.

Chenguang Wang, Mu Li, and Alexander J Smola. 2019. Language models with transformers. *arXiv preprint arXiv:1904.09408*.

Xinyi Wang, Hieu Pham, Zihang Dai, and Graham Neubig. 2018. SwitchOut: An efficient data augmentation algorithm for neural machine translation. In *EMNLP*.

Alex Warstadt, Alicia Parrish, Haokun Liu, Anhad Mohananey, Wei Peng, Sheng-Fu Wang, and Samuel R Bowman. 2020. BLiMP: The benchmark of linguistic minimal pairs for English. *TACL*.

Alex Warstadt, Amanpreet Singh, and Samuel R Bowman. 2019. Neural network acceptability judgments. *TACL*, 7:625–641.

Shinji Watanabe, Takaaki Hori, Shigeki Karita, Tomoki Hayashi, Jiro Nishitoba, Yuya Unno, Nelson Enrique Yalta Soplin, Jahn Heymann, Matthew Wiesner, Nanxin Chen, et al. 2018. ESPnet: End-to-end speech processing toolkit. In *INTERSPEECH*.

Thomas Wolf, L Debut, V Sanh, J Chaumond, C Delangue, A Moi, P Cistac, T Rault, R Louf, M Funtowicz, et al. 2019. HuggingFace's transformers: State-of-the-art natural language processing. *arXiv preprint arXiv:1910.03771*.

Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, et al. 2016. Google's neural machine translation system: Bridging the gap between human and machine translation. *arXiv preprint arXiv:1609.08144*.

Deyi Xiong, Min Zhang, and Haizhou Li. 2011. Enhancing language models in statistical machine translation with backward n-grams and mutual information triggers. In *ACL*.

Jiacheng Yang, Mingxuan Wang, Hao Zhou, Chengqi Zhao, Yong Yu, Weinan Zhang, and Lei Li. 2019a. Towards making the most of BERT in neural machine translation. *arXiv preprint arXiv:1908.05672*.

Yilin Yang, Liang Huang, and Mingbo Ma. 2018. Breaking the beam search curse: A study of (re-) scoring methods and stopping criteria for neural machine translation. In *EMNLP*.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R Salakhutdinov, and Quoc V Le. 2019b. XLNet: Generalized autoregressive pretraining for language understanding. In *NeurIPS*.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. 2020. BERTScore: Evaluating text generation with BERT. In *ICLR*.

A Experiment details

A.1 Language models

Implementation. English BERT, M-BERT, GPT-2, and RoBERTa models were served, adapted, and finetuned via the GluonNLP toolkit (Guo et al., 2020). German BERT and XLM models were served via HuggingFace's Transformers toolkit (Wolf et al., 2019). We release a reference implementation (a language model scoring package) for our work at https://github.com/awslabs/mlm-scoring.

Training. When adapting to a corpus we continue the training scheme for BERT, i.e., MLM + next-sentence prediction (Devlin et al., 2019), on the new dataset only, until the training loss converges. We still perform warmup at adaptation time (ratio of 0.01), but continue to use batches of 256 sequences of contiguous sentences, each with length up to 512.

Scoring. For BERT, M-BERT, and RoBERTa we prepend and append [CLS], [SEP] tokens. For GPT-2 we prepend and append <|endoftext|>, the default tokens for unconditional generation, as we found this outperformed other initial conditions (e.g., a preceding "##."). For XLM we prepend and append </s> (prepending <s> is more proper, but this is due to a bug in HuggingFace Transformer's XLMTokenizer that we will fix; changes in results should be negligible). When computing (pseudo-)perplexity (Section 2.3), these special tokens' conditional probabilities are not included, nor are they counted for token or word counts during length normalization.

N-best rescoring. We follow the log-linear model in Section 3.1 with its hyperparameter λ, i.e., weighted addition of (M)LM scores with sequence-to-sequence scores. When interpolating MLMs with GPT-2 there is also a hyperparamter γ (Appendix D). We do grid search on (λ, γ) with increments 0.05, 0.1 for the best weights on the development set towards downstream WER or BLEU, then evaluate on the corresponding test set. In the case of ties, we choose the largest λ or γ.

A.2 Automatic speech recognition

We use the LibriSpeech corpus (Panayotov et al., 2015) for our experiments. To adapt BERT we use the provided 800M-word text-only data, processing using Kaldi to match the normalized, downloadable corpus[2] but *with sentences in their original order* (instead of alphabetically), to match the long-context training regime of our language models. Our LibriSpeech-only BERT base model was trained on this corpus using GluonNLP's recipe, for 1.5M steps.

We take pre-existing 100-best lists shared via e-mail communication (Shin et al., 2019), which were produced by ESPnet (Watanabe et al., 2018) on LibriSpeech's dev and test sets. The ESPnet model was the sequence-to-sequence BLSTMP model in the *librispeech/asr1* recipe, except with 5 layers and a beam size of 100.

For speech corpora, to alleviate some of the domain shift from BERT's original written corpora, we appended "." at the end of utterances during adaptation, and appended "." to all hypotheses before subword tokenization, masking, and token/word counting.

A.3 Neural machine translation

Our pretrained model[3] is the base Transformer on WMT 2014 English-German (Vaswani et al., 2017) trained using GluonNLP's scripts/machine_translation. Evaluation and N-best rescoring was on the 3003-sentence test set via --full --bleu 13a --beam_size 100.

We consider 5 low-resource directions from the TED Talks dataset (Qi et al., 2018): Arabic (ar), Galician (gl), and Slovak (sk) to English; and English to Arabic, German (de), languages which were considered in Aharoni et al. (2019). We also include a more popular benchmark, English to Vietnamese (vi) from the IWSLT '15 evaluation campaign[4] (Cettolo et al., 2015). These give a breadth of English-source and English-target pairs and include a right-to-left language; more importantly, the three non-English targets are covered by the 15-language XLMs (Conneau and Lample, 2019).

Our models are also described as baselines in a dedicated work (Nguyen and Salazar, 2019). They are base Transformers with 6 layers, 8 heads, an 8k BPE vocabulary, and dropout of 0.3, except for gl→en where we use 4 layers, 4 heads, 3k BPE, and a dropout of 0.4 due to its significantly smaller

[2] https://www.openslr.org/resources/11/librispeech-lm-norm.txt.gz

[3] http://apache-mxnet.s3-accelerate.dualstack.amazonaws.com/gluon/models/transformer_en_de_512_WMT2014-e25287c5.zip

[4] https://nlp.stanford.edu/projects/nmt/

size. We use a warmup of 8k steps and the default hyperparameters (Vaswani et al., 2017). We apply GNMT length normalization (Wu et al., 2016) with $\alpha = 0.6$ to the sequence-to-sequence log-scores, and $\alpha = 1.0$ to the PLL scores (motivation is given in Section 4.3), with respect to their chosen tokenization's lengths. We compute tokenized BLEU via *multi-bleu.perl* from Moses[5] to compare with past works on these datasets.

B BERT as a generative model

In their published version (Wang and Cho, 2019), the authors claimed that BERT is a Markov random field language model (MRF-LM) where $\{\mathbf{w}_t\}_{t=1}^{|\mathbf{W}|}$ are categorical random variables (over the vocabulary) in a fully-connected graph X. They define a *potential* over cliques of X such that all partial-graph potentials are $\exp(0) = 1$ and the full-graph potential is $\exp \sum_{t=1}^{|\mathbf{W}|} \log \phi_t(X)$, where $\log \phi_t(X)$ is the logit corresponding to $\log P_{\text{MLM}}(\boldsymbol{w}_t \mid \boldsymbol{W}_{\setminus t})$ (although in their formulation, one could include the softmax into the feature function f_θ and take $\log \phi_t(X) = \text{PLL}(X)$ exactly).

Abusing notation, we write $\boldsymbol{W}$ interchangeably with its graph X. An MRF defined in this way would give the joint distribution:

$$P_{\text{MLM}}(\boldsymbol{W}) = \frac{1}{Z} \prod_{t=1}^{|\boldsymbol{W}|} \phi_t(\boldsymbol{W}) = \frac{1}{Z} \exp \text{PLL}(\boldsymbol{W}),$$

where Z is the partition function

$$Z = \sum_{\boldsymbol{W}'} \prod_{t=1}^{|\boldsymbol{W}|} \phi_t(\boldsymbol{W}') = \sum_{\boldsymbol{W}'} \exp \text{PLL}(\boldsymbol{W}'),$$

making this a valid distribution by normalizing over all sequences of the same length.

One then hopes to say that $\log P_{\text{MLM}}(\boldsymbol{w}_t \mid \boldsymbol{W}_{\setminus t})$ is the conditional distribution of this MRF. However, their erratum[6] notes this is not the case, as $\mathbf{w}_t$ would be affected by other log-potentials as well.

In practice, one could instead *a priori* make the modeling assumption

$$g(\boldsymbol{W}) = P_{\text{MLM}}(\boldsymbol{W}) := \frac{1}{Z} \exp \text{PLL}(\boldsymbol{W}),$$

as done in the work on bi-RNNLMs (Chen et al., 2017). They *choose* to model the distribution of sentences as a product-of-experts $\boldsymbol{w}_t \mid \boldsymbol{W}_{\setminus t}$, whose parameters are shared via the underlying bi-RNN.

Suppose one had access to this "normalized MLM probability". In the log-linear setting (Section 3.1), we get

$$\begin{aligned} &\log P_{\text{S2S}}(\mathbf{W} \mid \boldsymbol{X}) + \lambda \log g(\mathbf{W}) \\ &\quad = \cdots + \lambda \log \left(\frac{1}{Z} \exp \text{PLL}(\boldsymbol{W}) \right) \\ &\quad = \cdots + \lambda \, \text{PLL}(\boldsymbol{W}) - \lambda \log Z. \end{aligned}$$

For a fixed λ and Z (a property of the MLM), we see that $\lambda \log Z$ does not affect rank-ordering when taking the argmax to get the best hypothesis $\hat{\mathbf{W}}$. Hence, the heuristic interpolation enacted by λ is "the same" for normalized $\log P_{\text{LM}}$, unnormalized PLL, and our hypothetical $\log P_{\text{MLM}}$. The remaining issue is whether λ has the same effect for all lengths $|\mathbf{W}|$, which one mitigates by applying the correct length penalties to f and g (Section 4.3).

C Pseudo-perplexity and rescoring

We briefly examine the relationship between PPPL (Section 2.3) and metrics post-rescoring. We plot negative PLLs versus $|\boldsymbol{W}|$ and observe linearity, helping justify the simple average over length:

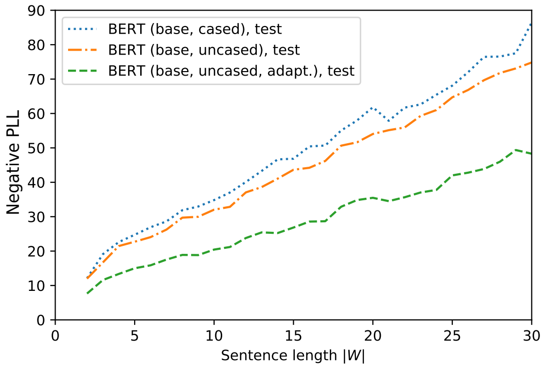

Figure 5: Negative pseudo-log-likelihood scores versus sentence length (in tokens) from BERT, averaged over LibriSpeech's test utterances of each length.

Note that in this section, we largely consider PPPLs normalized by *number of words* (PPPL$_\text{w}$), to improve comparability between different subword vocabularies. We see a good correspondence between PPPL$_\text{w}$ improvements and post-rescoring WER in Table 9 and BLEU in Table 10.

Thanks to this correspondence, one can compute a new pretrained model's PPPL on a small target-domain sample to quickly assess whether a model's rescoring could improve over a previous model.

[5] `https://statmt.org`

[6] "BERT has a Mouth and must Speak, but it is not an MRF" from `https://sites.google.com/site/deepernn/home/blog`

Model	**test**			
	clean		other	
	PPPL$_\text{w}$	WER	PPPL$_\text{w}$	WER
BERT (base, cased)	24.18	5.41	27.47	17.41
RoBERTa (base, cased)	21.85	5.25	24.54	17.18
BERT (large, cased)	17.49	5.25	19.59	16.97
BERT (base, uncased)	17.49	5.14	19.24	16.97
RoBERTa (large, cased)	14.78	5.05	16.23	16.79
BERT (base, Libri. only)	9.86	4.79	10.55	16.50
BERT (base, unc., adapt.)	**6.63**	**4.58**	**6.56**	**15.96**

Table 9: Word-normalized PPPL vs. WER on LibriSpeech after hypothesis rescoring for models with different token vocabularies. WERs are from Table 2 and Table 5.

Model	**dev**					
	ar→en		gl→en		sk→en	
	PPPL$_\text{w}$	BLEU	PPPL$_\text{w}$	BLEU	PPPL$_\text{w}$	BLEU
B-base	13.08	35.71	11.86	20.25	13.20	29.74
B-large	10.17	35.79	9.48	20.21	10.43	29.79
R-base	9.77	35.86	9.36	20.21	9.75	29.79
R-large	**6.26**	**36.02**	**6.08**	**20.44**	**6.29**	**30.05**

Table 10: Word-normalized PPPL vs. BLEU of cased BERT (B) and RoBERTa (R) on English gold sentences in the TED Talks corpus.

D Combining MLMs and GPT-2

We ask whether scores from a unidirectional LM are complementary with a masked LM for rescoring. When interpolating, we introduce γ such that:

$$\log g(\boldsymbol{W}) = (1-\gamma)\log P_{\text{LM}}(\boldsymbol{W}) + \gamma\,\text{PLL}(\boldsymbol{W}).$$

Our results are in Table 11:

Model	**test**		**+ GPT-2**	
	clean	other	clean	other
baseline (100-best)	7.26	20.37	5.30	17.26
BERT (large, cased)	5.25	16.97	5.03	16.80
RoBERTa (large, cased)	5.05	16.79	4.93	16.71
BERT (base, unc., adapt.)	4.58	15.96	**4.50**	**15.92**

Table 11: WERs on LibriSpeech after hypothesis rescoring, with and without interpolating with GPT-2 (345M, cased).

As the MLM gets stronger, the improvement from adding scores from GPT-2 goes to zero, suggesting that their roles overlap at the limit. However, unlike recent work (Shin et al., 2019) but like previous work (Chen et al., 2017), we found that interpolating with a unidirectional LM remained optimal, though our models are trained on different datasets and may introduce an ensembling effect.