Abstract

We rerank with scores from pretrained masked language models like BERT to improve ASR and NMT performance. These log-pseudolikelihood scores (LPLs) can outperform large, autoregressive language models (GPT-2) in out-of-the-box scoring. RoBERTa reduces WER by up to 30% relative on an end-to-end LibriSpeech system and adds up to +1.7 BLEU on state-of-the-art baselines for TED Talks low-resource pairs, with further gains from domain adaptation. In the multilingual setting, a single XLM can be used to rerank translation outputs in multiple languages. The numerical and qualitative properties of LPL scores suggest that LPLs capture sentence fluency better than autoregressive scores. Finally, we finetune BERT to estimate sentence LPLs without masking, enabling scoring in a single, non-recurrent inference pass.

Introduction

BERT (Devlin et al., 2019) and its improvements to natural language understanding have spurred a rapid succession of contextual language representations (Lample and Conneau, 2019; Yang et al., 2019b; Liu et al., 2019) which use larger datasets and more involved training schemes. Their success is attributed to their use of bidirectional context, often via the masked language model (MLM) objective. Here, a token $w_t$ is replaced with [MASK] and predicted using all past and future tokens $W_t := (\ldots, w_{t-1}, w_{t+1}, \ldots)$. In contrast, conventional language models predict $w_t$ using only past tokens $W_{\leq t} := (w_1, \ldots, w_{t-1})$.

Using pretrained contextual representations in sequence-to-sequence architectures involves integrating either their representations (Edunov et al., 2019) or weights (Yang et al., 2019a) into the encoder or decoder, then training from scratch. However, unidirectional models give log-probability estimates for a sentence $W$ via the chain rule:

$$\log P_{LM}(W) = \sum_{t=1}^{|W|} \log P_{LM}(w_t | W_{<t}).$$

These estimates naturally compose with scores from sequence-to-sequence models during decoding, leading to the continued use of language model rescoring in automatic speech recognition (ASR) (Tosnival et al., 2018; Irie et al., 2019) and neural machine translation (NMT) (Gulcehre et al., 2015; Stahlberg et al., 2018). Meanwhile, no prior work has evaluated using pretrained masked language models in this “plug-and-play” way.

Hence, we propose using log-pseudolikelihood scores (LPLs) (Wang and Cho, 2019) for rescoring in ASR and NMT, by way of sentence reranking. Each score is given by summing the conditional log-probabilities $\log P_{LM}(w_t | W_{<t})$ of each sentence token (Shin et al., 2019), as induced in BERT by replacing $w_t$ with [MASK]. Furthermore, we show that one can finetune BERT to compute LPLs in a single, non-recurrent inference pass. Our scoring and maskless finetuning scheme is shown in Figure 1.

Reranking with BERT competes with or even outperforms GPT-2 models (Radford et al., 2019), which are true language models of similar size but trained on more data. Gains scale with dataset and model size: RoBERTa (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 the TED Talks corpus. In the multilingual case, we find that a single 15-
language XLM (Lample and Conneau, 2019) can concurrently improve NMT into different target languages out-of-the-box.

Finally, we analyze the properties of LPLs and propose them as a starting point for future ranking or scoring schemes. Numerically, LPLs scale linearly with utterance length and exhibit robustness across reference translation pairs. Qualitatively, LPLs help disentangle fluency from adequacy, with positional cross-entropies visibly spiking at disfluencies due to domain mismatch.

2 Background

Let $X$ denote audio features or source text tokens, and let $W = (w_1, w_2, \ldots, w_{|W|})$ denote target text tokens. For non-end-to-end ASR and MT systems, having a separate model $P_{LM}(W)$ is motivated by the Bayes rule decomposition used to select the best hypothesis $W$ (Brown et al., 1993):

$$
W = \arg\max_W \log P(W | X) = \arg\max_W [\log P(X | W) + \log P(W)].
$$

2.1 The log-linear model

End-to-end ASR and NMT use encoder-decoder architectures that are trained discriminatively (Sutskever et al., 2014; Chan et al., 2016). Though less principled, many still adopt a log-linear model

$$
\hat{W} = \arg\max_W \log P(W | X) \\
\approx \arg\max_W [\log f(W, X) + \lambda \log g(W)]
$$

with learned functions $f, g$ and a hyperparameter $\lambda$, to good effect (Section 4.1). One often takes $f = P_{S2S}(W | X)$ as the sequence-to-sequence model and $g = P_{LM}(W)$ as the language model, and proceeds in one of two ways (possibly both):

Fusion. Autoregressive models allow $f$ and $g$ to be naturally indexed over time via the chain rule:

$$
\text{score} = \sum_t \log f_t(W, X) + \lambda \sum_t \log g_t(W)
= \sum_t \log P_{S2S}(w_t | W_{<t}, X) + \lambda \log P_{LM}(w_t | W_{<t})
= \sum_t \text{score}_t.
$$

Instead of performing $\arg\max_W$ at the end, beam search with width $K$ is often used to restrict to $K$ running sums at each time step. This general approach is known as fusion (Gulcehre et al., 2015).

Reranking. One computes $f(W, X)$ first, still using beam search to maintain the top $K$ hypotheses and scores. Then, $g(W)$ 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” \( W \) in its beam (further discussion in Appendix B), but we gain two advantages:

- **Length independence.** If \( g \) is non-recurrent, then \( g(W) \) can 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 rescoring (Li et al., 2019; Shin et al., 2019; Irie et al., 2019).

- **Scale independence.** Fusion requires correspondence between \( f_t \) and \( g_t \) at every \( t \), a property naturally met by autoregression. In reranking, \( f = \text{P}_{\text{S2S}} \) does not require \( g \) to decompose over time or to be a “true probability” at all; only that a suitable \( \lambda \) exists (the choice of log-linear versus linear is relevant here; see Chen et al. (2017b) for details).

These enables our use of log-pseudolikelihood scores for reranking and motivates our maskless finetuning approach.

### 2.2 Pseudolikelihood

Bidirectional contextual representations like BERT come at the expense of being a “true” language model \( P_L(W) \), as there appears no natural way to generate text (sampling) or produce likelihood scores (density estimation) from these models. This impedes their use in generative tasks, at best serving as initialization for encoder-decoder models (Edunov et al., 2019; Yang et al., 2019a) or unidirectional LMs (Wang et al., 2019).

Wang and Cho (2019) observed that BERT’s MLM objective corresponds to stochastic maximization of the pseudolikelihood estimate (MPLE) (Besag, 1975) on a training set \( \mathcal{W} \), where \( \{w_t \mid W\}_{t=1}^{|W|} \) are random variables in a fully-connected graph. The estimate is given by

\[
\mathcal{J}_{\text{PL}}(\Theta; \mathcal{W}) = \frac{1}{|\mathcal{W}|} \sum_{W \in \mathcal{W}} \text{LPL}(W),
\]

where LPL denotes the sentence-level pseudolikelihood estimate. BERT’s MPLE objective may endow it with similar performance to MLE, as MPLE is also (weakly) consistent and related via a relative entropy bound (Mozeika et al., 2014). We view LPL as a function of \( W \) and define the log-pseudolikelihood score:

\[
\text{LPL}(W) := \sum_{t=1}^{|W|} \log P_{\text{MLM}}(w_t \mid W_{\leq t}; \Theta).
\]

This motivated Wang and Cho (2019)’s use of Gibbs sampling to generate text with BERT, and led them to suggest (but not evaluate) LPLs as a proxy for density estimation. These summands are induced from BERT by keeping the positionwise softmax layer (the “MLM decoder”), replacing \( w_t \) with \([\text{MASK}]\), performing inference, then extracting \( w_t \)’s score at position \( t \).

Concurrently, Shin et al. (2019) sought to extend past work on future-conditional LMs in ASR (Section 5) with bidirectional self-attentive language models (bi-SANLMs). They train shallow models from scratch and use the same \([\text{MASK}]\) scoring method, but do not relate their work to pseudolikelihood, which provides a framework to explain their success and observed behaviors (Section 4). Experimentally, we extend their evaluations to pretrained models, to NMT, and to the multilingual setting (Section 3).

#### 2.3 \([\text{MASK}]\) less scoring

A caveat unaddressed in both works is that retrieving LPLs from an MLM requires a sentence copy for each position, which restores length-dependence in the number of inference passes (though unlike recurrence, these can be parallelized). Hence, we propose training a network \( q(W; \Theta_S) \) to match BERT’s LPLs without \([\text{MASK}]\) tokens. Specifically, we consider sentence-level regression towards the LPL sum:

\[
|\text{LPL}(W) - q(W; \Theta_S)|^2.
\]

To expedite training, we finetune \( q \) from the pretrained model \( P_{\text{MLM}} \) directly, replacing the softmax with a regression layer (Figure 1).

More generally, one could use any student model \( q \) as in knowledge distillation (Hinton et al., 2014). Here, the teacher gives individual token probabilities (\( T \) inference passes) while the student approximates their sum (one inference pass). This is also reminiscent of Oord et al. (2018), which distills an autoregressive WaveNet teacher to a parallel WaveNet student. Other \([\text{MASK}]\) less bidirectional models like XLNet (Yang et al., 2019b) could also give LPL estimates, though we leave this to future work.
3 Experiments and results

Further details can be found in Appendix A:

**LMs.** We rerank sequence-to-sequence hypotheses as in Section 2.1. Each hypothesis is given a log-likelihood score (uni-SANLM, GPT-2) or an LPL score (bi-SANLM, BERT, M-BERT, RoBERTa, XLM). We tune the LM weight $\lambda$ 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 Shin et al. (2019), who use an end-to-end, 5-layer BLSTM model 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 4.

**NMT.** Our 100-best hypotheses are from state-of-the-art subword base Transformer baselines on five low-resource pairs from the TED Talks corpus (Qi et al., 2018) and one from IWSLT 2015 (Cettolo et al., 2015). Length normalization (Wu et al., 2016) is applied to NMT ($\alpha = 0.6$) and LM ($\alpha = 1.0$) scores; we motivate this in Section 4.2.

### 3.1 Out-of-the-box (English)

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 reranking ASR outputs in Table 1.

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 training on less data. RoBERTa gives a relative WER reduction of 30% on LibriSpeech test-clean and 18% on test-other.

We repeat the same on English-target NMT in Table 2. As 100-best can be worse than 4-best due to the beam search curse (Yang et al., 2018; Murray and Chiang, 2018), we decode both beam sizes and find no systematic degradation in our models. Reranking 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 show conclusive improvements on Transformer encoder-decoder models via LM rescoring for the first time, despite only reranking; the most recent fusion work (Stahlberg et al., 2018) only used LSTM-based models.

### 3.2 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 optionally trained with a cross-lingual translation LM (TLM) objective (Lample and Conneau, 2019). Monolingual training was done on Wikipedia, which gives e.g., 6GB of German text. We also rerank with German BERT models trained on 16GB of text (similar to English

---

1 Four of our six baselines are described in a concurrent work (Nguyen and Salazar, 2019). The remainder (en→ar, en→de) use the same setup, chosen for coverage by XLM.

2 https://github.com/google-research/bert/blob/master/multilingual.md

3 https://github.com/dbmdz/german-bert
BERT); see Table 3.

| Model                  | IWSLT '15 | TED Talks |
|------------------------|-----------|-----------|
|                        | en→vi     | en→de     | en→ar     |
| # of training examples | 133k      | 167k      | 213k      |
| 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 3: Test BLEU scores for language pairs with non-English targets, after reranking. base* uses 1024 hidden dims. but only 8 heads instead.

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. We expect increase training data and model size will boost XLM performance, as with RoBERTa (large, cased) in Table 2.

### 3.3 Domain adaptation

Out-of-the-box reranking may be hampered 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 3.1. We rerank using in-domain models in Table 4:

| Model                  | dev | clean | other | test | clean | other |
|------------------------|-----|-------|-------|------|-------|-------|
| baseline (100-best)    | 7.17| 19.79 | 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 | 5.23  | 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 4: WERs on LibriSpeech after reranking. Baseline, SANLM, and oracle numbers are from Shin et al. (2019).

Using a BERT model trained only on the text corpus outperforms RoBERTa which is trained on far more data, underscoring the tradeoff 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 4.2 we find that these domain shifts can be visibly observed from position-wise LPL values.

The best results still come from adapting a pretrained model to the text corpus with further training (Appendix A). We train and adapt as in BERT, i.e., using large contiguous blocks of tokens. Shin et al. (2019) use shallow, 3-layer SANLMs but do utterance-wise training, which is slower but may reduce mismatch even further.

### 3.4 Finetuning without masking

We finetune BERT to produce scores without [MASK] tokens. For LibriSpeech we take one-fourth of the normalized corpus and keep sentences $|W| \leq 256$ for speed, score them with our adapted BERT base, then do sentence-level regression (Section 2.3). We train using Adam with a learning rate of $10^{-5}$ for 14 epochs then decay to $10^{-6}$ for 1 epoch (Table 5).

| 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 | 4.83| 15.73 |

Table 5: WERs on LibriSpeech using maskless scoring. Entries other than BERT are single-copy, non-recurrent passes.

Sentence-level finetuning degrades performance by +0.5-0.6 WER, leaving room for future improvement. This still outperforms GPT-2 (117M, cased), though this gap may 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 sequence-level knowledge distillation (Kim and Rush, 2016), for which our results are a naive performance bound.
Informally, $\text{LPL}(W)$ expresses how likely each token is given other tokens (self-consistency), while $\log P_{\text{LM}}(W)$ expresses the unconditional probability of a sentence, beginning with the costly unconditional term $P_{\text{LM}}(\text{Benedict})$. We demonstrate this effect numerically in Section 4.2.

| System                | Model          | Output sentence                                                                 |
|-----------------------|----------------|--------------------------------------------------------------------------------|
| LibriSpeech (dev-other) | Baseline      | clapping truth and jail ya in the mouth of the student is that building up or tearing down |
|                       | GPT-2          | clasping 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         | clasping truth into jail gagging the mouth of the student is that building up or tearing down |
| LibriSpeech (dev-other) | Baseline      | no preacher preach is more constructed |
|                       | GPT-2          | no preacher appears more constructed |
|                       | BERT (adapted) | no preacher preaches more constructed |
|                       | Target         | no preacher or priest is more constructive |
| 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 Sivities |
|                       | BERT           | I worked on a scientific status on the ACL, the Union by the Union of LiberCivís |
|                       | Target (en)    | I was working at the ACLU as the organization’s science advisor. |
| System                | Model          | Output sentence                                                                 |
|-----------------------|----------------|--------------------------------------------------------------------------------|
| Table 6: Examples of different top-1 hypotheses after reranking, with differences highlighted. GPT-2 and BERT both promote fluency, but GPT-2’s unconditional $P_{\text{LM}}(W)$ scores cause it to overweight common word sequences at the expense of adequacy.

4 Analysis

4.1 Fluency via self-consistency

Although end-to-end $P_{\text{S2S}}$ predict $W$ directly from $X$, interpolation with the unconditional LM score $g(W)$ remains widely used (Toshniwal et al., 2018). The fusion case where score$_t$ is log-linear (Section 2.1) is shallow fusion; using a learned dynamic combination is deep fusion (Gulcehre et al., 2015).

Later works introduced cold and simple fusion (Sriram et al., 2018; Stahlberg et al., 2018), which learn the language scorer first to surprising success. They argue that $g(W)$ expresses fluency; hence, fixing $g$ early allows $f(W, X)$ to focus its capacity on adequacy in representing the source, effectively disentangling the two.

Borrowing this intuition, we claim that taking $g = \text{LPL}(W)$ instead of $\log P_{\text{LM}}$ captures fluency independently of token frequency, which tempers degradation in adequacy. Consider a rare proper name like $W =$ “Benedict Cumberbatch”. It is a highly-fluent but low-probability bigram, as

$$
\log P_{\text{LM}}(W) = \log P_{\text{LM}}(\text{Benedict}) + \log P_{\text{LM}}(\text{Cumberbatch} | \text{Benedict}) \\
\ll \log P_{\text{MLM}}(\text{Benedict} | \text{Cumberbatch}) + \log P_{\text{MLM}}(\text{Cumberbatch} | \text{Benedict}) = \text{LPL}(W).
$$

Qualitatively, we examine where the baseline, GPT-2, and BERT gave different top-1 hypotheses (Table 6). In our ASR examples, GPT-2 restores fluency using common words:

- clapping truth and $\rightarrow$ class in truth and, no preacher preach $\rightarrow$ no preacher appears.

One can view this as an exacerbation of the rare-word problem due to overconfident logits (Nguyen and Chiang, 2018). Meanwhile, BERT only rewards self-consistency, which lets rarer but still-fluent words with better acoustic scores to persist:

- clapping truth and $\rightarrow$ clapping truth in, no preacher preach $\rightarrow$ no preacher preaches.

The former preserves the $p$ in the ground truth (clapping), while the latter preserves the second $pr$ in the ground truth (priest).

NMT is known to be highly fluent but inadequate (Tu et al., 2017), leading to over- and under-translation (Tu et al., 2016). In our gl→en example, LPLs temper this behavior while log-likelihood scores exacerbate it. GPT-2 reranks

**Union by the Union Sivities $\rightarrow$ Union by the Union by the Union Civís**

which is even more over-translated (Union repeated thrice) than the baseline. BERT prefers the more globally-fluent **Union by the Union of LiberCivís**, which also corrects the under-translation (i.e., omission) of Liber without being discouraged by the rare sequence LiberCivís.

4.2 Numerical properties of LPL

LPL’s numerical properties make it an ideal foundation for future ranking or scoring schemes.

---

### Table 6: Examples of different top-1 hypotheses after reranking, with differences highlighted. GPT-2 and BERT both promote fluency, but GPT-2’s unconditional $P_{\text{LM}}(W)$ scores cause it to overweight common word sequences at the expense of adequacy.
For example, given fixed $|W|$ one expects $-\log P_{\text{MLM}}(w_t \mid W_{<t})$ to be similar for all $t$, while $-\log P_{\text{LM}}(w_t \mid W_{<t})$ decreases as $t \to |W|$, as observed by Takahashi and Tanaka-Ishii (2018) in recurrent language models. We validate this with GPT-2 (Figure 2) and BERT (Figure 3). In particular, we see the outsized cost of the unconditional first unigram in Figure 2. These also explain Shin et al. (2019)'s observation that bi-SANLM was more robust than uni-SANLM at shorter and earlier positions; the difference is intrinsic to log-probabilities versus LPLs, and is not due to model or data size.

Figure 2: Cross-entropy (natural base) of $w_t \mid W_{<t}$ versus context length ($t - 1$) as given by GPT-2 models, averaged over LibriSpeech’s test utterances.

In Figure 4 we plot sentence-level LPLs and observe linearity as $|W| \to \infty$, with spikes from the last word and uncapitalized first word averaging out. This behavior motivates $\alpha = 1.0$ for the length penalty of LPLs, which corresponds to the asymptotically-linear $L_{\text{PLM}} = (5 + |W|)/(5 + 1)$. In contrast, autoregressive scores like $P_{\text{LM}}(W)$ integrate over the inverse power law curve in Figure 2. We speculate that this explains the effectiveness of Wu et al. (2016)’s length penalty ($\alpha = 0.6$), widely used in NMT baselines including ours, as there exists some $C$ such that

$$L_{\text{PLS2S}}(W) = (5 + |W|)^{0.6} \approx \int_0^{|W|} \frac{C}{(5 + x)^{0.4}} dx.$$ 

Finally, scores from a naive multilingual autoregressive LM can be sensitive on the amount of data presented per language. For example, a model trained with more English than Spanish text would give $P_{\text{LM}}(\text{Hey}) \gg P_{\text{LM}}(\text{Hola})$. This has motivated the use of initial language tokens (<en>, <es>) to condition multilingual sequence
models (Johnson et al., 2017). In contrast, LPLs leverage context from future same-language tokens. M-BERT uses this to justify training without language markers, as the above probabilities would be replaced by $P_{\text{MLM}}(\text{Hey} \mid \text{how are ...})$ and $P_{\text{MLM}}(\text{Hola} \mid \text{cómo estás ...})$.

We explore whether this makes LPLs robust across reference translations (e.g., casual speech ↔ casual speech). We plot length-normalized LPL estimates for our paired English and Vietnamese sentences using cased M-BERT (Figure 5). These averaged LPLs match remarkably well, as seen visually by our equivalent-axes plot and a correlation of $r = 0.45$. There is a slight shift upwards (more negative LPLs on the Vietnamese half), which may be due to shared wordpieces between Indo-European languages.

![Figure 5: M-BERT’s negative length-normalized LPLs ($\alpha = 1.0$) for the en→vi development set pairs.](image)

### 4.3 System combination

Given the different behaviors of LPL and LL scores, we explore whether ensembling the two can further improve performance. When interpolating we introduce $\gamma$ such that:

$$\log g(W) = (1 - \gamma) \log P_{\text{LL}}(W) + \gamma \log P_{\text{LPL}}(W).$$

Our results are in Table 7:

| Model               | test clean other | + GPT-2 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 7: WERs on LibriSpeech after reranking, with and without interpolating with GPT-2 (345M, cased).

As the LPL model gets stronger, the improvement from adding scores from GPT-2 goes to zero, suggesting that their roles overlap at the limit (Section 4.1). However, unlike Shin et al. (2019) but like Chen et al. (2017b), we found that interpolating with a unidirectional LM remained optimal, though our models are trained on different dataset and may introduce an ensembling effect.

### 4.4 Pseudoperplexity

We define pseudoperplexity (PPPL) analogously to perplexity, i.e., $\exp\left(-\frac{1}{|W_a|} \sum \log P_{\text{MLM}}(w_t \mid W_{a t})\right)$. To encourage future exploration, we briefly test the relationship between pseudoperplexity and downstream metrics. For comparability, we compute word-level PPPL, instead of dividing by the number of tokens $|W|$, we divide by the number of words. We see a mild correspondence between PPPL improvements and post-reranking WER and BLEU in Table 8 and Table 9.

| Model               | clean     | other     | clean     | other     |
|---------------------|-----------|-----------|-----------|-----------|
| BERT (base, cased)  | 24.18 5.41| 27.47 17.41|
| BERT (large, cased) | 17.49 5.25| 19.59 16.97|
| BERT (base, uncased)| 17.49 5.14| 19.24 16.97|
| + adaptation, 380k steps | 6.63 4.58| 6.56 15.96 |

Table 8: PPPL vs. WER on LibriSpeech after reranking.

| Model | dev | sk → en |
|-------|-----|---------|
| PPPL  | BLEU| PPPL    | BLEU    |
| B-base| 21.01 35.71 | 62.14 20.25 | 28.16 29.74 |
| B-large| 18.38 35.79 | 57.26 20.21 | 25.06 29.79 |
| R-base| 14.92 35.86 | 45.79 20.21 | 20.20 29.79 |
| R-large| 12.66 36.02 | 37.29 20.44 | 17.04 30.05 |

Table 9: PPPL vs. BLEU of cased BERT (B) and RoBERTa (R) on English gold sentences in the TED Talks corpus.

The midlines denote a change in tokenization, which can bias PPPL computation as defined above; this can be mitigated in the future by whole-word masking. In practice, we found that computing a new pretrained model’s PPPL on a small sample helped quickly assess whether reranking would be worthwhile over a previous model.

### 5 Related work

The closest works are Wang and Cho (2019) and Shin et al. (2019), whose experimental differences are outlined in Section 2.2. Neither works
consider the inference cost of masked reranking, which we address with our maskless scoring approach, or analyze LPL’s numerical properties.

**Future context.** Log-probabilities conditioned on past and future context have been used in machine translation (MT) (Finch and Sumita, 2009; Xiong et al., 2011) and perennially in automatic speech recognition (ASR) (Shi et al., 2013; Arisoy et al., 2015; Chen et al., 2017a,b) to positive effect. However, these concatenate forward and backward LMs; they do not model the interaction of $w_{<t}$ and $w_{>t}$ and are not LPLs (e.g., their cross-entropies in Figure 3 would be convex, not flat).

**Language model integration.** Beyond fusion and initialization from pretrained MLMs, monolingual pretraining has improved NMT performance (Ramachandran et al., 2017; Lample and Conneau, 2019). However, compositional integrations of language representation models remain prevalent, especially in ASR. Contemporary examples are the use of BERT’s finetuned scores for passage reranking (Nogueira and Cho, 2019) or ‘as-is’ cosine similarity scores from BERT to evaluate generated text (Zhang et al., 2019). For example, in decoder pretraining one may have no pretrained multilingual LMs, which are difficult to train (Ragni et al., 2016), but may already have finetuned BERT to a target language/domain. Fusion and reranking are not mutually exclusive with pretraining, although pretraining may be redundant in capturing fluency (Section 4.1).

6 Conclusion

We evaluated the effectiveness of reranking with pretrained masked language models for modern sequence-to-sequence models in both ASR and low-resource NMT. We found they can match or outperform reranking with comparable unidirectional language models. We attributed this to LPLs, namely their promotion of fluency via self-consistency instead of “likeliness”. Future work could include finding additional compositional uses of masked LMs, simplifying non-masked LPL computations, and using LPLs 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, pages 3874–3884.

Ebru Arisoy, Abhinav Sethy, Bhuvana Ramabhadran, and Stanley Chen. 2015. Bidirectional recurrent neural network language models for automatic speech recognition. In ICASSP, pages 5421–5425.

Julian Besag. 1975. Statistical Analysis of Non-Lattice Data. The Statistician, 24(3):179.

Peter F. Brown, Stephen A. Della Pietra, Vincent J. 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 Catteni, 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, pages 4960–4964.

X. Chen, X. Liu, A. Ragni, Y. Wang, and M. J. F. Gales. 2017a. Future word contexts in neural network language models. In ASRU, pages 97–103.

X Chen, Anton Ragni, X Liu, and Mark J F Gales. 2017b. Investigating Bidirectional Recurrent Neural Network Language Models for Speech Recognition. In INTERSPEECH, pages 269–273.

Zihang Dai, Zhilin Yang, Yiming Yang, Jaime Carbonell, Quoc V. Le, and Ruslan Salakhutdinov. 2019. Transformer-XL: Attentional Language Models Beyond a Fixed-Length Context. ACL, 1:2978–2988.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In NAACL-HLT, pages 4171–4186.

Andrey Edunov, Alexei Baevski, and Michael Auli. 2019. Pre-trained language model representations for language generation. In NAACL-HLT, pages 4052–4059.

Andrew Finch and Eiichiro Sumita. 2009. Bidirectional Phrase-based Statistical Machine Translation. In EMNLP, pages 1124–1132.

Caglar Gulcehre, Orhan Firat, Kelvin Xu, Kyunghyun Cho, Loic Barrault, Hui-Chi Lin, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2015. On Using Monolingual Corpora in Neural Machine Translation. arXiv.
Jian Guo, He He, Tong He, Leonard Lausen, Mu Li, Haibin Lin, Xingjian Shi, Chenguang Wang, Junyuan Xie, Sheng Zha, Aston Zhang, Hang Zhang, Zhi Zhang, Zhongyue Zhang, and Shuai Zheng. 2019. GluonCV and GluonNLP: Deep Learning in Computer Vision and Natural Language Processing. arXiv.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. 2014. Distilling the Knowledge in a Neural Network. Deep Learning Workshop, NeurIPS.

Kazuki Irie, Albert Zeyer, Ralf Schlüter, and Hermann Ney. 2019. Language Modeling with Deep Transformers. In INTERSPEECH, pages 3905–3909.

Melvin Johnson, Mike Schuster, Quoc V. Le, Maxim Krikun, Yonghui Wu, Zhifeng Chen, Nikhil Thorat, Fernanda Viégas, Martin Wattenberg, Greg Corrado, and Jeffrey Dean. 2017. Google’s Multilingual Neural Machine Translation System: Enabling Zero-Shot Translation. In TACL, 5:339–351.

Yoon Kim and Alexander M. Rush. 2016. Sequence-Level Knowledge Distillation. In EMNLP, pages 1317–1327.

Guillaume Lample and Alexis Conneau. 2019. Cross-lingual Language Model Pretraining. arXiv.

Jason Li, Vitaly Lavrukhin, Boris Ginsburg, Ryan Leary, Oleksii Kuchaiev, Jonathan M. Cohen, Huyen Nguyen, and Ravi Teja Gadge. 2019. Jasper: An End-to-End Convolutional Neural Acoustic Model. In INTERSPEECH, pages 71–75.

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.

Elman Mansimov, Alex Wang, and Kyunghyun Cho. 2019. A Generalized Framework of Sequence Generation with Application to Undirected Sequence Models. arXiv.

Alexander Mozeika, Onur Dikmen, and Joonas Piili. 2014. Consistent inference of a general model using the pseudolikelihood method. Physical Review E, 90(1):010101.

Kenton Murray and David Chiang. 2018. Correcting Length Bias in Neural Machine Translation. In WMT, pages 212–223.

Graham Neubig and Junjie Hu. 2018. Rapid Adaptation of Neural Machine Translation to New Languages. EMNLP, pages 875–880.

Toan Nguyen and David Chiang. 2018. Improving Lexical Choice in Neural Machine Translation. In NAACL-HLT, pages 334–343.

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.

Aaron Oord, Yazhe Li, Igor Babuschkin, Karen Simonyan, Oriol Vinyals, Koray Kavukcuoglu, George Driessche, Edward Lockhart, Luis Cobo, Florian Stimberg, Norman Casagrande, Dominik Grewe, Seb Noury, Sander Dieleman, Erich Elsen, Nal Kalchbrenner, Heiga Zen, Alex Graves, Helen King, Tom Walters, Dan Belov, and Demis Hassabis. 2018. Parallel WaveNet: Fast High-Fidelity Speech Synthesis. In ICML, pages 3918–3926.

Vassil Panayotov, Guoguo Chen, Daniel Povey, and Sanjeev Khudanpur. 2015. Librispeech: An ASR corpus based on public domain audio books. In ICASSP, pages 5206–5210.

Ye Qi, Devendra Sachan, Matthieu Felix, Sarguna Ramanabhan, and Graham Neubig. 2018. When and Why Are Pre-Trained Word Embeddings Useful for Neural Machine Translation? In NAACL-HLT, pages 529–535.

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 J F Gales, and Kate M Knill. 2016. Multi-Language Neural Network Language Models. In INTERSPEECH, pages 3042–3046.

Prajit Ramachandran, Peter Liu, and Quoc Le. 2017. Unsupervised Pretraining for Sequence to Sequence Learning. In EMNLP, pages 383–391.

Yangyang Shi, Martha Larson, Pascal Wiggers, and Catholijn M Jonker. 2013. Exploiting the succeeding words in recurrent neural network language models. In INTERSPEECH, pages 632–636.

Joongbo Shin, Yoonhyung Lee, and Kyomin Jung. 2019. Effective Sentence Scoring Method using Bidirectional Language Model for Speech Recognition. arXiv.

Anuroop Sriram, Heewoo Jun, Sanjeev Satheesh, and Adam Coates. 2018. Cold Fusion: Training Seq2Seq Models Together with Language Models. In INTERSPEECH, pages 387–391.

Felix Stahlberg, James Cross, and Veselin Stoyanov. 2018. Simple Fusion: Return of the Language Model. In WMT, pages 204–211.

Qing Sun, Stefan Lee, and Dhruv Batra. 2017. Bidirectional beam search: Forward-backward inference in neural sequence models for fill-in-the-blank image captioning. In CVPR, volume 2017-Janua, pages 7215–7223.

Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. 2014. Sequence to Sequence Learning with Neural Networks. In NeurIPS, pages 3104–3112.
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*, pages 369–375.

Zhaopeng Tu, Yang Liu, Lifeng Shang, Xiaohua Liu, and Hang Li. 2017. Neural machine translation with reconstruction. In *AAAI*, pages 3097–3103.

Zhaopeng Tu, Zhengdong Lu, Liu Yang, Xiaohua Liu, and Hang Li. 2016. Modeling coverage for neural machine translation. In *ACL*, volume 1, pages 76–85.

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*, pages 5998–6008.

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*.

Xinyi Wang, Hieu Pham, Zihang Dai, and Graham Neubig. 2018. SwitchOut: an Efficient Data Augmentation Algorithm for Neural Machine Translation. In *EMNLP*, pages 856–861.

Yuguang Wang, Shanbo Cheng, Liyang Jiang, Jiajun Yang, Wei Chen, Muze Li, Lin Shi, Yanfeng Wang, and Hongtao Yang. 2017. Sogou Neural Machine Translation Systems for WMT17. In *WMT*, pages 410–415.

Shinji Watanabe, Takaaki Hori, Shigeki Karita, Tomoki Hayashi, Jiro Nishitoba, Yuya Unno, Nelson Enrique Yalta Soplin, Jahn Heymann, Matthew Wiesner, Nanxin Chen, Adithya Renduchintala, and Tsubasa Ochiai. 2018. ESPnet: End-to-End Speech Processing Toolkit. In *INTERSPEECH*, pages 2207–2211.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, and Jamie Brew. 2019. HuggingFace’s Transformers: State-of-the-art Natural Language Processing. *arXiv*.

Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V. Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, Jeff Klingner, Apurva Shah, Melvin Johnson, Xiaohong Liu, Łukasz Kaiser, Stephan Gouws, Yoshikiyo Kato, Taku Kudo, Hiroyuki Kazawa, Keith Stevens, George Kurian, Nishant Patil, Wei Wang, Cliff Young, Jason Smith, Jason Riesa, Alex Rudnick, Oriol Vinyals, Greg Corrado, Macduff Hughes, and Jeffrey Dean. 2016. Google’s Neural Machine Translation System: Bridging the Gap between Human and Machine Translation. *arXiv*.

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*, pages 1288–1297.

Jiacheng Yang, Mingxuan Wang, Hao Zhou, Chengqi Zhao, Yong Yu, Weiman Zhang, and Lei Li. 2019a. Towards Making the Most of BERT in Neural Machine Translation. *arXiv*.

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*, pages 3054–3059.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Ruslan Salakhutdinov, and Quoc V. Le. 2019b. XLNet: Generalized Autoregressive Pre-training for Language Understanding. In *NeurIPS*.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2019. BERTScore: Evaluating Text Generation with BERT. *arXiv*.

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., 2019). German BERT and XLM models were served via HuggingFace’s Transformers toolkit (Wolf et al., 2019).

**Training.** When adapting to a corpus we continue the training scheme, i.e., MLM + next-sentence prediction (Devlin et al., 2019) for BERT, 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 contiguous sequences of length 512.

**Scoring.** For BERT, M-BERT, and RoBERTa we prepend and append `[CLS]`, `[SEP]` tokens. For GPT-2 we prepend and append `<|endoftext|>`, finding 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 XLMTokenizer that we will fix; changes in results should be negligible). These special tokens are not masked for prediction, nor are they included in token or word counts for (pseudo)perplexity (Section 4.4).
**Reranking.** We follow the log-linear model in Section 2.1 with its hyperparameter $\lambda$. We do grid search on $(\lambda, \gamma)$ 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 case of ties, we choose the largest $\lambda$ or $\gamma$.

### 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 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.

From Shin et al. (2019) we take their 100-best lists (shared via e-mail communication) produced by ESPnet (Watanabe et al., 2018) on LibriSpeech’s dev and test sets. The ESPnet model was the sequence-to-sequence BLSTM model in the `librispeech/asr1` recipe, except with 5 layers and a beam size of 100. For speech corpora we use “..” as sentence-boundary marker during adaptation, and append “..” to all hypotheses before tokenization, masking, and token/word counts.

### A.3 Neural machine translation

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 (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 (Lample and Conneau, 2019).

Our models are Transformers with 6 layers, 8 heads, 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 size. We use a warmup of 8k steps and use 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 LPL scores (motivation is given in Section 4.2), with respect to their chosen tokenization’s lengths. We compute tokenized BLEU via `multi-bleu.perl` from Moses to compare with past works on these datasets.

### B Shallow fusion and reranking

Shallow fusion sums the two models’ scores at each time step before proceeding with a beam width of $K$. This can be understood as truncating the support of one’s distribution at each time step (Kim and Rush, 2016):

$$ f_t g_t^\lambda \mapsto \frac{f_t g_t^\lambda}{\sum_{w_t \in T_K(\prod_{s \leq t} f_s g_s^\lambda)} f_t g_t^\lambda} $$

where $T_K(v)$ denotes the top-$K$ elements of the distribution $v$. In contrast, reranking applies the beam width of $K$ to $f$ only, i.e.,

$$ f_t \mapsto \frac{f_t}{\sum_{w_t \in T_K(\prod_{s \leq t} f_s)} f_t} $$

per timestep, then multiplies the result with $g = \prod g_t$. Though fusion seems more effective (one could view it as regularizing per time step, rather than at the end) and is more prevalent when log-linear interpolation occurs, reranking has found occasional effective use in earlier end-to-end ASR and NMT models (Chan et al., 2016; Wang et al., 2017). To make the two equivalent, one would need the top $K$ of $f_t$ to map to itself when acted upon by $g_t^\lambda$, an idea related to disentanglement (Section 4.1) we defer to future work.

Combined with the above, we see that unidirectional beam search can be problematic, especially with the disproportionate cost of early unconditional unigrams (Section 4.2) in log-likelihood scoring. Some mitigations include bidirectional beam search (Sun et al., 2017), though a more general non-monotonic decoding framework may ultimately be best (Mansimov et al., 2019).