InforMask: Unsupervised Informative Masking for Language Model Pretraining

Nafis Sadeq*, **Canwen Xu***, **Julian McAuley**
University of California, San Diego
{nsadeq,cxu,jmcauley}@ucsd.edu

Abstract

Masked language modeling is widely used for pretraining large language models for natural language understanding (NLU). However, random masking is suboptimal, allocating an equal masking rate for all tokens. In this paper, we propose InforMask, a new unsupervised masking strategy for training masked language models. InforMask exploits Pointwise Mutual Information (PMI) to select the most informative tokens to mask. We further propose two optimizations for InforMask to improve its efficiency. With a one-off preprocessing step, InforMask outperforms random masking and previously proposed masking strategies on the factual recall benchmark LAMA and the question answering benchmark SQuAD v1 and v2.[1]

1 Introduction

Masked Language Modeling (MLM) is widely used for training language models (Devlin et al., 2019; Liu et al., 2019; Lewis et al., 2020; Raffel et al., 2020). MLM randomly selects a portion of tokens from a text sample and replaces them with a special mask token (e.g., [MASK]). However, random masking has a few drawbacks — it sometimes produces masks that are too easy to guess, providing a small loss that is inefficient for training; some randomly masked tokens can be guessed with only local cues (Joshi et al., 2020); all tokens have an identical probability to be masked, while (e.g.) named entities are more important and need special attention (Sun et al., 2019; Levine et al., 2021).

In this paper, we propose a new strategy for choosing tokens to mask in text samples. We aim to select words with the most information that can benefit the language model, especially for knowledge-intense tasks. To tackle this challenge, we propose *InforMask*, an unsupervised informative masking strategy for language model pretraining. First, we introduce *Informative Relevance*, a metric based on Pointwise Mutual Information (PMI, Fano, 1961) to measure the quality of a masking choice. Optimizing this measure ensures the informativeness of the masked token while maintaining a moderate difficulty for the model to predict the masked tokens. This metric is based on the statistical analysis of the corpus, which does not require any supervision or external resource.

However, maximizing the total Informative Relevance of a text sample with multiple masks can be computationally challenging. Thus, we propose a sample-and-score algorithm to reduce the time complexity of masking and diversify the patterns in the output. An example is shown in Figure 1. For training a language model with more epochs, we can further accelerate the masking process by only running the algorithm once as a preprocessing step and assigning a token-specific masking rate for each token according to their masking frequency in the corpus, to approximate the masking decisions of the sample-and-score algorithm. After this one-off preprocessing step, masking can be as fast as the original random masking without any further overhead, which can be desirable for large-scale distributed language model training of many epochs.

To verify the effectiveness of our proposed method, we conduct extensive experiments on two knowledge-intense tasks — factual recall and question answering. On the factual recall benchmark LAMA (Petroni et al., 2019), InforMask outperforms other masking strategies by a large margin. Also, our base-size model, InformBERT, trained with the same corpus and epochs as BERT (Devlin et al., 2019) outperforms BERT-base on question answering benchmark SQuAD (Rajpurkar et al., 2016, 2018). Notably, on the LAMA benchmark, InformBERT outperforms BERT and

*Equal contribution.

[1] The code and model checkpoints are available at https://github.com/NafisSadeq/InforMask.

Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing, pages 5866–5878
December 7-11, 2022 ©2022 Association for Computational Linguistics

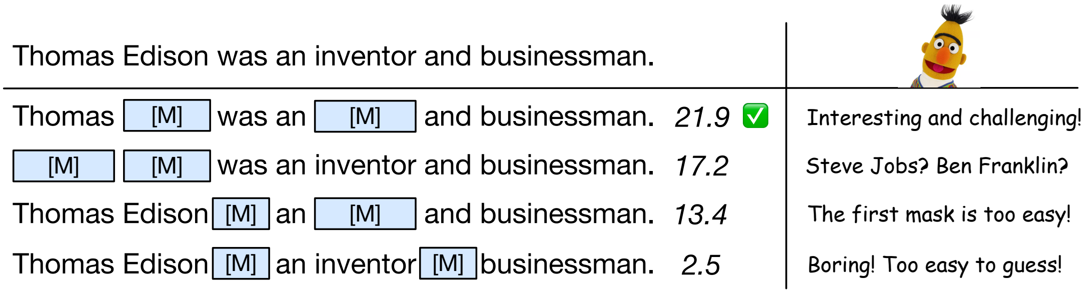

Thomas Edison was an inventor and businessman.
Thomas [M] was an [M] and businessman. 21.9 ✅
Interesting and challenging!
[M] [M] was an inventor and businessman. 17.2
Steve Jobs? Ben Franklin?
Thomas Edison [M] an [M] and businessman. 13.4
The first mask is too easy!
Thomas Edison [M] an inventor [M] businessman. 2.5
Boring! Too easy to guess!

Figure 1: The informative scores of randomly sampled masking candidates ($s = 4$). [M] denotes the masked tokens. The pretraining objective of the masked language model (MLM) is to predict the masked tokens based on the context.

RoBERTa (Liu et al., 2019) models that have 3× parameters and 10× corpus size.

To summarize, our contributions are as follows:

- We propose InforMask, an informative masking strategy for language model pretraining that does not require extra supervision or external resource.
- We pretrain and release InformBERT, a base-size English BERT model that substantially outperforms BERT and RoBERTa on the factual recall benchmark LAMA despite having much fewer parameters and less training data. InformBERT also achieves competitive results on the question answering datasets SQuAD v1 and v2.

2 Related Work

Random Masking For pretraining Transformer (Vaswani et al., 2017) based language models such as BERT (Devlin et al., 2019), a portion of the tokens is randomly chosen to be masked to set up the masked language model (MLM) objective. Prior studies have commonly used a masking rate of 15% (Devlin et al., 2019; Joshi et al., 2020; Levine et al., 2021; Sun et al., 2019; Lan et al., 2020; He et al., 2021), while some recent studies argue that masking rate of 15% may be a limitation (Clark et al., 2020) and the pretraining process may benefit from increasing the masking rate to 40% (Wettig et al., 2022). However, random masking is not an ideal choice for learning factual and commonsense knowledge. Words that have high informative value may be masked less frequently compared to (e.g.) stop words, given their frequencies in the corpus.

Span Masking Although random masking is effective for pretraining a language model, some prior works have attempted to optimize the masking procedure. Joshi et al. (2020) propose SpanBERT where they show improved performance on downstream NLP tasks by masking a span of words instead of individual tokens. They randomly select the starting point of a span, then sample a span size from a geometric distribution and mask the selected span. They continue to mask spans until the target masking rate is met. This paper suggests masking spans instead of single words can prevent the model from predicting masked words by only looking at local cues. However, this masking strategy inevitably reduces the modeling between the words in a span, etc., Mount-Fuji, Mona-Lisa, which may hinder its performance in knowledge-intense tasks.

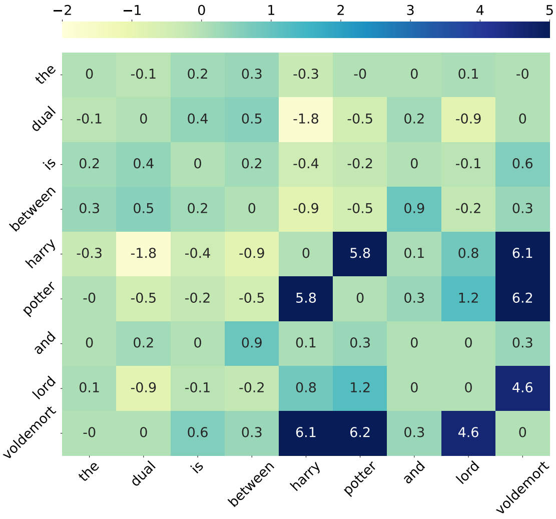
Entity-based Masking Baidu-ERNIE (Sun et al., 2019) introduces an informed masking strategy where a span containing named entities will be masked. This approach shows improvement compared to random masking but requires prior knowledge regarding named entities. Similarly, Guu et al. (2020) propose Salient Span Masking where a span corresponding to a unique entity will be masked. They rely on an off-the-shelf named entity recognition (NER) system to identify entity names. LUKE (Yamada et al., 2020) exploits an annotated entity corpus to explicitly mark out the named entities in the pretraining corpus, and masks non-entity words and named entities separately.

PMI Masking Levine et al. (2021) propose a masking strategy based on Pointwise Mutual Information (PMI, Fano, 1961), where a span of up to five words can be masked based on the joint PMI of the span of words. PMI-Masking is an adaption of SpanBERT (Joshi et al., 2020) where meaningful spans are masked instead of random ones. However, PMI-Masking only considers correlated spans and fails to focus on unigram named entities. This may lead to suboptimal performance on knowledge intense tasks (details in Section 4.2). In our proposed method, we exploit PMI to determine the informative value of tokens to encourage more efficient training and improve performance on knowledge-intense tasks.

Knowledge-Enhanced LMs KnowBERT (Peters et al., 2019) shows that factual recall performance in BERT can be improved significantly by embedding knowledge bases into additional layers of the model. Tsinghua-ERNIE (Zhang et al., 2019) exploits a similar approach that injects knowledge graphs into the language model during pretraining. KEPLER (Wang et al., 2021) uses a knowledge base to jointly optimizes the knowledge embedding loss and MLM loss on a general corpus, to improve the knowledge capacity of the language model. Similar ideas are also explored in K-BERT (Liu et al., 2020) and CoLAKE (Sun et al., 2020). CokeBERT (Su et al., 2021) demonstrates that incorporating embeddings for dynamic knowledge context can be more effective than incorporating static knowledge graphs. Other works have attempted to incorporate knowledge in the form of lexical relation (Lauscher et al., 2020), word sense (Levine et al., 2020), syntax (Bai et al., 2021), and parts-of-speech (POS) tags (Ke et al., 2020). However, a high-quality knowledge base is expensive to construct and not available for many languages. Different from these methods, our method is fully unsupervised and does not rely on any external resource.

3 Methodology

InforMask aims to make masking decisions more 'informative'. Since not all words are equally rich in information (Levine et al., 2021), we aim to automatically identify more important tokens (e.g., named entities) and increase their probability to be masked while preserving the factual hints to recover them. On the other hand, we would like to reduce the frequency of masking stop words. Stop words are naturally common in the corpus and they can be important for learning the syntax and structure of a sentence. However, masked stop words can be too easy for a language model to predict, especially in later stages of LM pretraining. Thus, properly reducing the masking frequency of stop words can improve both the efficiency and performance of the model.



	the	dual	is	between	harry	potter	and	lord	voldemort
the	0	-0.1	0.2	0.3	-0.3	-0	0	0.1	-0
dual	-0.1	0	0.4	0.5	-1.8	-0.5	0.2	-0.9	0
is	0.2	0.4	0	0.2	-0.4	-0.2	0	-0.1	0.6
between	0.3	0.5	0.2	0	-0.9	-0.5	0.9	-0.2	0.3
harry	-0.3	-1.8	-0.4	-0.9	0	5.8	0.1	0.8	6.1
potter	-0	-0.5	-0.2	-0.5	5.8	0	0.3	1.2	6.2
and	0	0.2	0	0.9	0.1	0.3	0	0	0.3
lord	0.1	-0.9	-0.1	-0.2	0.8	1.2	0	0	4.6
voldemort	-0	0	0.6	0.3	6.1	6.2	0.3	4.6	0

Figure 2: The PMI matrix of the words in the sentence 'The dual is between Harry Potter and Lord Voldemort.'

3.1 Informative Relevance

To generate highly informative masking decisions for a sentence, we introduce a new concept, namely Informative Relevance. Informative Relevance is used to measure how relevant a masked word is to the unmasked words so that it can be meaningful and predictable. The Informative Relevance of a word is calculated by summing up the Pointwise Mutual Information (PMI, Fano, 1961) between the masked word and all unmasked words in the sentence. PMI between two words w_1 and w_2 represents how 'surprising' is the co-occurrence between two words, accounting for their own probabilities. Formally, the PMI of the combination $w_1 w_2$ is defined as:

$$\mathrm{pmi}(w_1, w_2) = \log \frac{p(w_1, w_2)}{p(w_1)p(w_2)} \tag{1}$$

The PMI matrix is calculated corpus-wise. Note that instead of using bigrams (i.e., two words have to be next to each other), we consider the skip-gram co-occurrence within a window. The window size is selected in a way that enables sentence-level co-occurrence to be considered as well as local co-occurrence.

Maximizing the Informative Relevance enables the model to better memorize knowledge and focus on more informative words. Since Informative Relevance is calculated between a masked word and the unmasked words, it also encourages hints to be preserved so that the model can reasonably

Algorithm 1 InforMask Algorithm

```
$D \leftarrow$ Set of text
$s \leftarrow$ Size of randomly sampled candidates
$F_i^d \leftarrow$ Informative score for $i$-th masking candidate for text $d$
for $d \in D$ do
    for $i = 1, 2, \ldots, s$ do
        Generate $i$-th masking candidate for $d$
        $M_i^d \leftarrow$ Masked Tokens
        $U_i^d \leftarrow$ Unmasked Tokens
        $F_i^d \leftarrow 0$
        for $w_1 \in M_i^d$ do
            for $w_2 \in U_i^d$ do
                $F_i^d = F_i^d + \mathrm{pmi}(w_1, w_2)$
            end for
        end for
    end for
    Choose candidate with maximum $F_i^d$
end for
```

Data Subset	#Relations	#Samples
ConceptNet	1	29774
Squad	1	305
GoogleRE	3	4994
TREx	41	34032
Total	46	69105

Table 1: Statistics of LAMA (Petroni et al., 2019).

Dataset	SQuAD v1	SQuAD v2
#Examples	108k	151k
#Negative Examples	0	54k
#Articles	536	505

Table 2: Statistics of SQuAD v1 and v2 (Rajpurkar et al., 2016, 2018).

guess the masked words. As shown in Figure 2, the words inside a named entity have a high PMI (e.g., 'Harry-Potter' and 'Lord-Voldemort') while the two closely related entities also show a high PMI (e.g., Harry-Voldemort). Thus, if we are asked to mask one word, we would mask 'Voldemort' since it has the highest Informative Relevance with the remaining words (by summing up the last row or column).

3.2 Scoring Masking Candidates

One text sample can have multiple masks. Thus, we define the informative score of a masking decision as the sum of the Informative Relevance of each masked token. However, given the PMI matrix, finding the best k words to mask (i.e., the masking decision with the highest informative score) in a sentence of n words is time-consuming. Iterating all possibilities has time complexity $O(C_n^k)$. By converting it to a minimum cut problem, the time complexity can be reduced to $O(n^2 \log n)$ (Stoer and Wagner, 1997), which is still prohibitive in practice.

Therefore, we propose to sample s random masking candidates and then rank them by calculating their informative scores. As shown in Figure 1, we randomly generate four masking candidates and rank them by their informative scores. We select the candidate with the highest score. This allows us to make a masking decision with time complexity $O(kn)$. Random sampling also introduces more diverse patterns for masking, which could help training of language models and prevent overfitting. This process is illustrated in Algorithm 1.

3.3 Token-Specific Masking Rates

Algorithm 1 is already usable by processing the input text on the fly. However, to avoid overfitting, masking should change across epochs. This means we have to run Algorithm 1 every epoch, creating a bottleneck for pretraining. To address this efficiency issue, we use token-specific masking rates to approximate the masking decisions of InforMask. Specifically, we generate masks for a corpus using Algorithm 1, and then count the frequency of each token in the vocabulary to be masked as their token-specific masking rates. Note that in this way, Algorithm 1 is only executed once, as a prepossessing step. Furthermore, we can use a small portion of the corpus to calculate the token-specific masking rates, making it even faster.[2] After this, we can perform random masking, except that every token has its own masking rate.

4 Experiments

4.1 Experimental Settings

Pretraining Corpus Following BERT (Devlin et al., 2019), we use the Wikipedia and Book Corpus datasets available from Hugging Face (Lhoest et al., 2021). The corpus contains ∼3.3B tokens. To be consistent with BERT, we use an overall masking rate of 15%. The PMI matrix is calculated

[2] For the Wikipedia corpus, the average rate of change for token-specific masking rates falls below 0.8% after processing only 1% of the corpus.

	Model	#Param.	Corpus Size	Epochs	LAMA (Petroni et al., 2019)				
					ConceptNet	Squad	GoogleRE	TREx	Overall
(a)	Random (2019)	125M	16 GB	3	0.091	0.124	0.396	0.582	0.549
	Span (2020)	125M	16 GB	3	0.056	0.102	0.377	0.524	0.495
	PMI (2021)	125M	16 GB	3	0.075	0.115	0.396	0.552	0.522
	InforMask	125M	16 GB	3	**0.109**	**0.133**	**0.410**	**0.627**	**0.591**
(b)	BERT-base	110M	16 GB	40	0.191	0.229	0.340	0.587	0.553
	BERT-large	340M	16 GB	40	0.218	0.284	0.354	0.621	0.585
	RoBERTa-base	125M	160 GB	40	0.223	0.307	0.423	0.630	0.592
	RoBERTa-large	355M	160 GB	40	**0.260**	0.329	0.435	0.672	0.632
	InformBERT	125M	16 GB	40	0.201	**0.384**	**0.509**	**0.739**	**0.698**

Table 3: Performance of different masking strategies and models on LAMA (Petroni et al., 2019). (a) We compare the models trained with different masking strategies for 3 epochs. (b) We compare InformBERT, a BERT model trained with InforMask for 40 epochs with BERT and RoBERTa models.

	Model	#Param.	Corpus Size	Epochs	SQuAD v1		SQuAD v2	
					F1	EM	F1	EM
(a)	Random (2019)	125M	16 GB	3	79.08	69.44	66.48	63.15
	Span (2020)	125M	16 GB	3	78.88	69.04	64.95	61.38
	PMI (2021)	125M	16 GB	3	80.31	70.98	66.25	62.82
	InforMask	125M	16 GB	3	**80.47**	**71.41**	**67.29**	**63.90**
(b)	BERT-base	110M	16 GB	40	81.07	88.52	72.35	75.75
	InformBERT	125M	16 GB	40	**81.22**	**88.61**	**72.71**	**75.86**

Table 4: Performance on SQuAD v1 and v2 (Rajpurkar et al., 2016, 2018) development set.

on the Wikipedia corpus, with a size of 100k × 100k. Word co-occurrence statistics are computed with a window size of 11. We set the candidate sampling size per document s to 30. It takes ∼4 hours to preprocess and generate token-specific masking rates on a 16-core CPU server with 256 GB RAM.

Evaluation Benchmarks To evaluate different masking strategies, we use the LAMA benchmark (Petroni et al., 2019) to test the knowledge of the models. LAMA is a probe for analyzing the factual and commonsense knowledge contained in pretrained language models. Thus, it is suitable for evaluating the knowledge learned during pretraining. LAMA has around 70,000 factual probing samples across 46 factual relations. A summary of the benchmark is shown in Table 1. We use Mean Reciprocal Rank (MRR) as the metric for factual recall performance.

In addition to the knowledge probing task, we also conduct experiments on real-world question answering datasets, which requires commonsense knowledge as well. We conduct experiments on SQuAD v1 and v2 (Rajpurkar et al., 2016, 2018) and report the F1 and Exact Match (EM) scores on the development set. The statistics of the benchmark are shown in Table 2. We provide additional results on GLUE (Wang et al., 2018) benchmark in Appendix C.

Baselines We compare InforMask in two settings: **(a)** We use the same tokenizer and hyperparameters to pretrain BERT random masking (Devlin et al., 2019), SpanBERT (Joshi et al., 2020) and PMI-Masking (Levine et al., 2021) for 3 epochs. The choice of 3 epochs is according to our limited computational budget. **(b)** We continue training InforMask until 40 epochs. The 40-epoch model is denoted as *InformBERT*. We compare InformBERT to BERT-base (Devlin et al., 2019), which is trained with the same corpus for 40 epochs as well. We also include results of BERT-large and RoBERTa for reference, though they are either larger in size or trained with more data and thus are not directly comparable.

Training Details Our implementation is based on Hugging Face Transformers (Wolf et al., 2020). We train the baselines and our model with 16 Nvidia

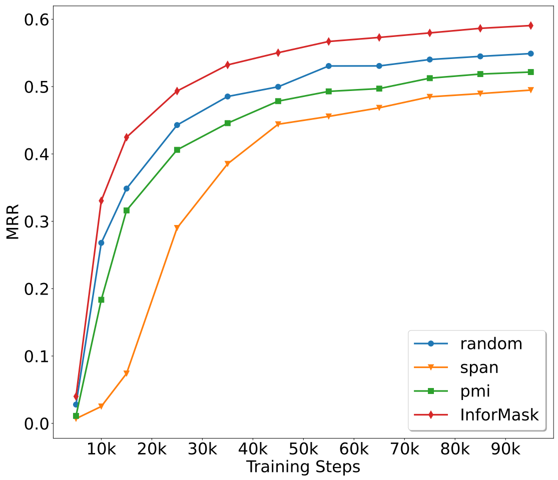


Figure 3: Macro average MRR of different masking strategies on LAMA, evaluated every 10k steps.

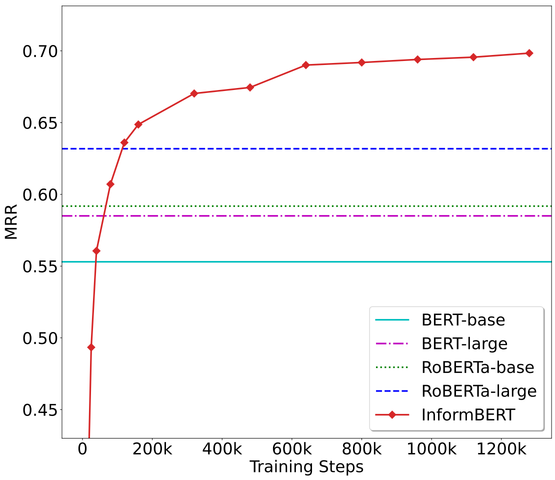


Figure 4: Performance of InformBERT for the full pretraining process. It achieves comparable performance with BERT-base after 40k training steps and even RoBERTa-large after 120k training steps.

V100 32GB GPU. For our model and all baselines trained, we use a fixed vocabulary size of 50,265. The model architecture is a base-size BERT model, with 12 Transformer layers with 12 attention heads. The hidden size is set to 768. The overall batch size is 256. We use an AdamW optimizer (Loshchilov and Hutter, 2019) with a learning rate of 5e-5. Note that we do not perform any hyperparameter searching or tuning for any model (including InformBERT) given our limited computational budget.

4.2 Experimental Results

Impact of Masking Strategies We conduct a fair comparison among different masking strategies, using the same tokenization and hyperparameters. As shown in Table 3(a), InforMask outperforms other masking strategies by a large margin on all subsets of LAMA (Petroni et al., 2019). As shown in Table 4(a), on both SQuAD v1 and v2 (Rajpurkar et al., 2016, 2018), InforMask outperforms other masking strategies. Notably, PMI-Masking achieves higher performance on SQuAD while underperforming random masking on LAMA (to be detailed shortly) but our InforMask achieves better results on both of them.

Also, we compare our 40-epoch InformBERT model with BERT and RoBERTa models. As shown in Table 3(b), InformBERT outperforms the BERT model trained with the same epochs and corpus by 0.145 overall. It also achieves higher performance than RoBERTa-base, despite being trained with 10% of RoBERTa's corpus size. To our surprise, it also outperforms both BERT-large and RoBERTa-large, with only 1/3 parameters. The breakdown of performance for each relation can be found in Appendix A. Moreover, InformBERT outperforms BERT-base for fine-tuning on SQuAD v1 and v2, demonstrating its capability for downstream question answering, as shown in Table 4(b).

Training Dynamics As shown in Figure 3, InforMask demonstrates an outstanding training efficiency. InforMask outperforms other masking strategies from the beginning of the training process and keeps the lead through the training. Notably, span masking and PMI-Masking underperform random masking, indicating their inability on the knowledge-intense task. Span masking also significantly underperforms other masking strategies in the early stage of pretraining, suggesting it may take longer to train the model. For the entire pretraining process, as shown in Figure 4, the model trained with InforMask outperforms BERT and RoBERTa with fewer than ∼15% of the training steps, verifying the efficiency of our masking strategy.

Impact on Stop Words and Entities As shown in Figure 5, without explicitly specifying the stop words, InforMask can identify the stop words and reduce their probability to be masked. InforMask can also automatically increase the masking probability of named entities. The average masking probability of named entities is 0.25 with a standard deviation of 0.07, while the overall masking

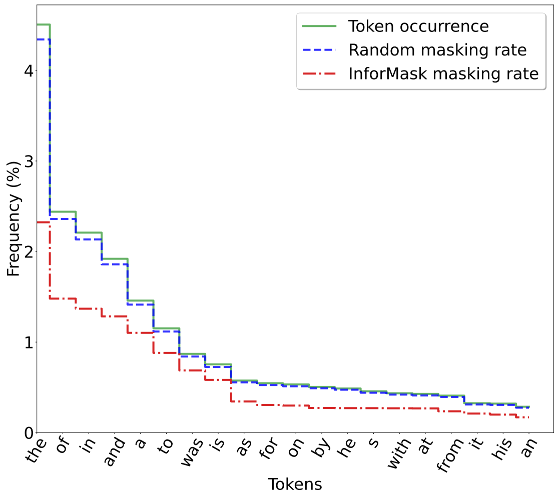


Figure 5: Frequency of common stop words and their corresponding masking rates by InforMask.

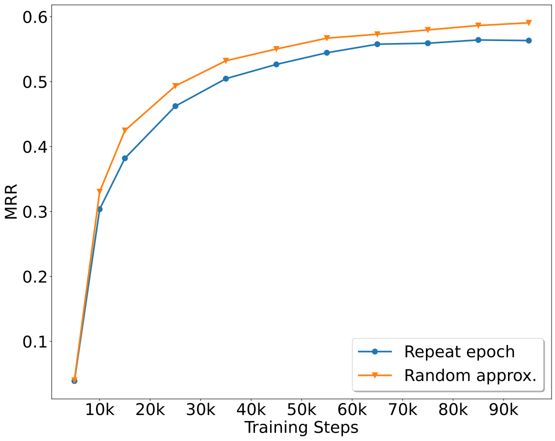


Figure 6: Comparison between looping the same data and using token-specific masking rate to approximate the masking decisions. The models are trained for 3 epochs.

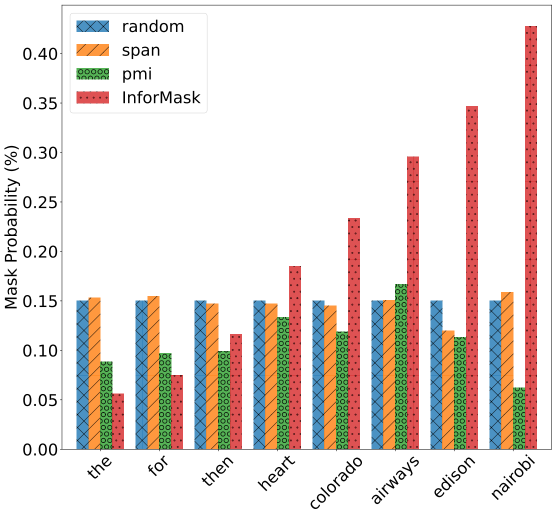


Figure 7: Masking rate of tokens according to different masking policies.

probability of all tokens is around 0.15.[3] This allows the model to focus on more important tokens and maintain an appropriate difficulty of prediction, facilitating the pretraining process.

Impact of Token-Specific Masking Rates As mentioned before, the use of token-specific masking rate can enormously save time and RAM for data processing, as spending hours of processing for each epoch can be infeasible and becomes a bottleneck for distributed training. Another possible solution is to loop the same masked data for every epoch. Thus, we conduct an experiment to compare the two solutions: approximation and repetition. Note that for simplicity, the token-specific masking rate is applied from the first epoch. As shown in Figure 6, our approximation strategy keeps outperforming the repetition strategy even in the first epoch. As we analyze, this can be attributed to the more diverse patterns introduced during the approximation. Also, the performance of the model trained with the repetition strategy converges or even slightly declines after 60k training steps while the performance of the model trained with approximation keeps increasing.

InforMask vs. PMI Masking PMI Masking (Levine et al., 2021) uses PMI to mask a span of correlated tokens. A named entity often constitutes a correlated span and therefore, is more likely to be masked in PMI-Masking. As mentioned before, we observe that PMI-Masking performs worse than random masking on LAMA (see Figure 3).

To investigate the reason, we compute the individual masking rates of some tokens according to each masking policy. As shown in Figure 7, we can see that PMI-Masking increases the masking rate of tokens that are part of correlated spans. However, it decreases the masking rate of tokens that are not within any correlated span, even if that token is a named entity. Consider the token 'Airways' for example. This token may be part of a correlated span such as 'British Airways' or 'Qatar Airways'. PMI-Masking, therefore, increases the masking rate of this token compared to random

[3] We use an off-the-shelf named entity recognition system to verify the effectiveness of our approach only. It is not a necessary component of the proposed system.

Query	Ground Truth	InformBERT		RoBERTa-base	
		Prediction	Score	Prediction	Score
Antoine Coypel was born in `[MASK]`.	paris	france **paris** haiti	0.09 0.08 0.04	montreal toronto **paris**	0.12 0.03 0.03
SpeedWeek is an American television program on `[MASK]`.	espn	**espn** nbc mtv	0.20 0.10 0.09	cbs cnbc spike	0.18 0.13 0.10
Phil Harrison is a corporate vice president of `[MASK]`.	microsoft	**microsoft** ibm motorola	0.20 0.15 0.05	intel ibm **microsoft**	0.06 0.05 0.03
Laurent Casanova was a `[MASK]` politician.	french	**french** canadian haitian	0.43 0.32 0.05	young **french** successful	0.13 0.09 0.04
The chief administrators of the church are `[MASK]`.	bishops	**bishops** priests appointed	0.13 0.07 0.06	men christians women	0.17 0.09 0.08

Table 5: Some examples of InformBERT and RoBERTa-base predictions on LAMA (Petroni et al., 2019). We show the queries and the ground-truth answers with the model predictions. We only show the top-3 predictions made by each model.

masking. On the other hand, the tokens 'Colorado' and 'Nairobi', which are unigram named entities, are less likely to be masked, compared to random masking. Given that the overall masking rate is fixed and PMI-Masking favors correlated spans, the masking rates of 'Colorado' and 'Nairobi' inevitably get lower. This can be the reason behind PMI-Masking's failure.

In contrast, InforMask uses PMI to compute the individual Informative Relevance of tokens. It can increase the masking rate of tokens with high informative saliency, regardless of whether they are part of a correlated span or not. This helps InforMask achieve superior factual recall performance.

4.3 Case Study

Table 5 shows the example knowledge probes and answers produced by InformBERT and RoBERTa. For the query 'SpeedWeek is an American television program on [MASK].', RoBERTa is unable to produce the correct answer in the top-3 predictions. But InformBERT correctly predicts 'ESPN' to be the top candidate. Similarly, InformBERT correctly predicts the answer 'bishops' for the query 'The chief administrators of the church are [MASK].' RoBERTa is unable to predict the answer and produces more generic words such as 'men', 'women', and 'Christians'.

We summarize the errors into two notable categories. They are relevant for all the models involved, not just InformBERT. First, we observe that many errors involve rare named entities. Some named entities are less frequent so the model is unable to learn anything useful about them, or they occur so rarely that they do not even appear in the language model vocabulary. We found that around 19% of the errors made by our model on the LAMA benchmark is associated with out-of-vocabulary tokens. Second, it is challenging for a language model to predict the granularity of the fact being asked or distinguish it from an alternate fact that may hold for a query. For the example query 'Antoine Coypel was born in [MASK].', the LAMA dataset has only one true label 'Paris'. In this example, InformBERT prefers the name of the country ('France') over the name of a city ('Paris'). This confusion is related to the granularity of location and both answers can be considered correct. However, it is being classified as an error because the labels in the test set are not comprehensive.

Another type of confusion can be found for RoBERTa with the query 'Laurent Casanova was a [MASK] politician.'. The model is trying to decide whether to use the adjective 'young', 'French', or 'successful'. In theory, these three adjectives may be valid simultaneously for the same entity. It can be challenging for the language model to pick the expected one in the context. We include more examples of knowledge probes with InformBERT in Appendix B.

5 Conclusion

In this work, we propose InforMask, an unsupervised masking policy that masks tokens based on their informativeness. InforMask achieves superior performance in knowledge-intense tasks including factual recall and question answering. We explore the impact of different masking strategies on learning factual and commonsense knowledge from pretraining and analyze why previously proposed masking techniques are suboptimal. For future work, we would like to scale up the pretraining and explore more factors for knowledge acquisition during unsupervised text pretraining.

Limitations

We conduct experiments to compare InforMask to several prior works on better masking strategies by training them for 3 epochs. We also compare a fully trained InformBERT-base model to BERT and RoBERTa. However, one limitation of our paper is due to our limited computational budget, we are not able to scale the experiments for larger model size, larger corpus, or compare all baselines under the full pretraining setting. Also, our InformBERT model is arguably suboptimal, with a relatively small batch size and no hyperparameter tuning or search at all.

Ethics Statement

Similar to BERT or RoBERTa, our model may contain social biases that preexist in the training corpus. Thus, we do not anticipate any major ethical concerns in addition to those identified in language models (Bender et al., 2021). However, to the best of our knowledge, there is no research on the impact of masking strategies on social biases, which could be an interesting and important direction for future research.

Acknowledgements

We would like to thank the anonymous reviewers for their insightful comments. This project is partly supported by NSF Award #1750063.

References

Jiangang Bai, Yujing Wang, Yiren Chen, Yaming Yang, Jing Bai, Jing Yu, and Yunhai Tong. 2021. Syntax-bert: Improving pre-trained transformers with syntax trees. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume, EACL 2021, Online, April 19 - 23, 2021*, pages 3011–3020. Association for Computational Linguistics.

Emily M. Bender, Timnit Gebru, Angelina McMillan-Major, and Shmargaret Shmitchell. 2021. On the dangers of stochastic parrots: Can language models be too big? In *FAccT*, pages 610–623. ACM.

Kevin Clark, Minh-Thang Luong, Quoc V. Le, and Christopher D. Manning. 2020. ELECTRA: pre-training text encoders as discriminators rather than generators. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 4171–4186. Association for Computational Linguistics.

Robert M Fano. 1961. Transmission of information: A statistical theory of communications. *American Journal of Physics*, 29(11):793–794.

Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, and Mingwei Chang. 2020. Retrieval augmented language model pre-training. In *International Conference on Machine Learning*, pages 3929–3938. PMLR.

Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. 2021. Deberta: decoding-enhanced bert with disentangled attention. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net.

Mandar Joshi, Danqi Chen, Yinhan Liu, Daniel S Weld, Luke Zettlemoyer, and Omer Levy. 2020. Spanbert: Improving pre-training by representing and predicting spans. *Transactions of the Association for Computational Linguistics*, 8:64–77.

Pei Ke, Haozhe Ji, Siyang Liu, Xiaoyan Zhu, and Minlie Huang. 2020. Sentilare: Sentiment-aware language representation learning with linguistic knowledge. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pages 6975–6988. Association for Computational Linguistics.

Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. 2020. ALBERT: A lite BERT for self-supervised learning of language representations. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net.

Anne Lauscher, Olga Majewska, Leonardo F. R. Ribeiro, Iryna Gurevych, Nikolai Rozanov, and Goran Glavas. 2020. Common sense or world knowledge? investigating adapter-based knowledge injection into pretrained transformers. *CoRR*, abs/2005.11787.

Yoav Levine, Barak Lenz, Or Dagan, Ori Ram, Dan Padnos, Or Sharir, Shai Shalev-Shwartz, Amnon Shashua, and Yoav Shoham. 2020. Sensebert: Driving some sense into BERT. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 4656–4667. Association for Computational Linguistics.

Yoav Levine, Barak Lenz, Opher Lieber, Omri Abend, Kevin Leyton-Brown, Moshe Tennenholtz, and Yoav Shoham. 2021. Pmi-masking: Principled masking of correlated spans. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 7871–7880. Association for Computational Linguistics.

Quentin Lhoest, Albert Villanova del Moral, Yacine Jernite, Abhishek Thakur, Patrick von Platen, Suraj Patil, Julien Chaumond, Mariama Drame, Julien Plu, Lewis Tunstall, et al. 2021. Datasets: A community library for natural language processing. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 175–184, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Weijie Liu, Peng Zhou, Zhe Zhao, Zhiruo Wang, Qi Ju, Haotang Deng, and Ping Wang. 2020. K-BERT: enabling language representation with knowledge graph. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 2901–2908. AAAI Press.

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. *CoRR*, abs/1907.11692.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net.

Matthew E. Peters, Mark Neumann, Robert L. Logan IV, Roy Schwartz, Vidur Joshi, Sameer Singh, and Noah A. Smith. 2019. Knowledge enhanced contextual word representations. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*, pages 43–54. Association for Computational Linguistics.

Fabio Petroni, Tim Rocktäschel, Sebastian Riedel, Patrick S. H. Lewis, Anton Bakhtin, Yuxiang Wu, and Alexander H. Miller. 2019. Language models as knowledge bases? In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*, pages 2463–2473. Association for Computational Linguistics.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67.

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

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

Mechthild Stoer and Frank Wagner. 1997. A simple min-cut algorithm. *Journal of the ACM (JACM)*, 44(4):585–591.

Yusheng Su, Xu Han, Zhengyan Zhang, Yankai Lin, Peng Li, Zhiyuan Liu, Jie Zhou, and Maosong Sun. 2021. Cokebert: Contextual knowledge selection and embedding towards enhanced pre-trained language models. *AI Open*, 2:127–134.

Tianxiang Sun, Yunfan Shao, Xipeng Qiu, Qipeng Guo, Yaru Hu, Xuanjing Huang, and Zheng Zhang. 2020. Colake: Contextualized language and knowledge embedding. In *Proceedings of the 28th International Conference on Computational Linguistics, COLING 2020, Barcelona, Spain (Online), December 8-13, 2020*, pages 3660–3670. International Committee on Computational Linguistics.

Yu Sun, Shuohuan Wang, Yu-Kun Li, Shikun Feng, Xuyi Chen, Han Zhang, Xin Tian, Danxiang Zhu, Hao Tian, and Hua Wu. 2019. ERNIE: enhanced representation through knowledge integration. *CoRR*, abs/1904.09223.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 5998–6008.

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

Xiaozhi Wang, Tianyu Gao, Zhaocheng Zhu, Zhengyan Zhang, Zhiyuan Liu, Juanzi Li, and Jian Tang. 2021. Kepler: A unified model for knowledge embedding and pre-trained language representation. *Transactions of the Association for Computational Linguistics*, 9:176–194.

Alexander Wettig, Tianyu Gao, Zexuan Zhong, and Danqi Chen. 2022. Should you mask 15% in masked language modeling? *CoRR*, abs/2202.08005.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Rémi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander M. Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations, EMNLP 2020 - Demos, Online, November 16-20, 2020*, pages 38–45. Association for Computational Linguistics.

Ikuya Yamada, Akari Asai, Hiroyuki Shindo, Hideaki Takeda, and Yuji Matsumoto. 2020. LUKE: Deep contextualized entity representations with entity-aware self-attention. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6442–6454, Online. Association for Computational Linguistics.

Zhengyan Zhang, Xu Han, Zhiyuan Liu, Xin Jiang, Maosong Sun, and Qun Liu. 2019. ERNIE: enhanced language representation with informative entities. In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 1441–1451. Association for Computational Linguistics.

A Performance Breakdown on LAMA

Subset	Relation	BERT-base	BERT-large	RoBERTa-base	RoBERTa-large	InformBERT
ConceptNet	test	0.191	0.218	0.223	**0.260**	0.201
GoogleRE	dateOfBirth	0.108	0.115	0.092	0.108	**0.122**
GoogleRE	placeOfBirth	0.475	0.493	0.610	0.612	**0.732**
GoogleRE	placeOfDeath	0.388	0.403	0.528	0.582	**0.607**
Squad	test	0.229	0.284	0.307	0.329	**0.384**
TREx	P1001	0.786	0.817	0.810	0.846	**0.881**
TREx	P101	0.453	0.499	0.307	0.380	**0.507**
TREx	P103	0.842	0.876	0.841	0.857	**0.907**
TREx	P106	0.656	0.675	0.540	0.599	**0.674**
TREx	P108	0.584	0.596	0.658	**0.725**	0.704
TREx	P127	0.546	0.570	0.661	0.688	**0.743**
TREx	P1303	0.387	0.442	0.233	0.277	**0.445**
TREx	P131	0.650	0.685	0.742	0.778	**0.867**
TREx	P136	0.621	0.666	0.557	0.596	**0.675**
TREx	P1376	0.730	0.768	0.631	0.630	**0.840**
TREx	P138	0.509	0.533	0.515	0.548	**0.742**
TREx	P140	0.606	0.674	0.668	0.728	**0.751**
TREx	P1412	0.777	0.801	0.799	0.824	**0.860**
TREx	P159	0.468	0.486	0.660	0.701	**0.789**
TREx	P170	0.860	0.886	0.878	0.908	**0.928**
TREx	P176	0.687	0.731	0.717	0.770	**0.777**
TREx	P178	0.631	0.683	0.711	**0.744**	0.721
TREx	P19	0.424	0.441	0.620	0.652	**0.760**
TREx	P190	0.267	0.312	0.486	0.542	**0.662**
TREx	P20	0.516	0.553	0.675	0.703	**0.791**
TREx	P264	0.273	0.300	0.003	0.005	**0.380**
TREx	P27	0.767	0.796	0.853	0.884	**0.895**
TREx	P276	0.549	0.577	0.646	0.682	**0.824**
TREx	P279	0.554	0.589	0.512	0.560	**0.594**
TREx	P30	0.832	0.868	0.845	0.896	**0.918**
TREx	P31	0.650	0.665	0.597	0.631	**0.652**
TREx	P36	0.425	0.447	0.484	0.511	**0.758**
TREx	P361	0.554	0.596	0.442	0.480	**0.607**
TREx	P364	0.738	0.767	0.661	0.704	**0.811**
TREx	P37	0.734	0.766	0.711	0.743	**0.788**
TREx	P39	0.615	0.647	0.501	0.550	**0.636**
TREx	P407	0.648	0.705	0.665	**0.710**	0.695
TREx	P413	0.480	0.501	0.508	**0.564**	0.508
TREx	P449	0.470	0.473	0.652	0.685	**0.735**
TREx	P463	0.676	0.692	0.641	0.683	**0.736**
TREx	P47	0.532	0.582	0.606	0.628	**0.860**
TREx	P495	0.707	0.737	0.805	**0.855**	0.823
TREx	P527	0.499	0.571	0.492	**0.585**	0.575
TREx	P530	0.448	0.493	0.740	**0.812**	0.802
TREx	P740	0.343	0.369	0.672	0.715	**0.731**
TREx	P937	0.554	0.587	0.720	0.741	**0.797**

Table 6: Relation by relation performance comparison on LAMA (Petroni et al., 2019).

B More LAMA Examples

Query	Ground truth	Top predictions (with confidence)
Communicating is for gaining [M].	knowledge	knowledge(0.22), information(0.09), insight(0.04)
Competing against someone requires a desire to [M].	win	compete(0.35), win(0.25), fight(0.08)
Going on the stage is for performing an [M].	act	act(0.65), opera(0.2), improvisation(0.02)
Playing is a way to [M] social skills.	learn	learn(0.22), develop(0.15), improve(0.14)
Gallagher was born on 14 December 1978 in [M] .	scotland	ireland(0.25), scotland(0.07), dublin(0.05)
Crisp died at her home in [M], Arizona .	phoenix	tucson(0.34), phoenix(0.12), prescott(0.12)
Frank Marion died in 1963 in [M], Connecticut .	stamford	hartford(0.12), stamford(0.1), middletown(0.1)
Mattingly died in 1951 in [M], Kentucky .	louisville	louisville(0.4), lexington(0.14), ashland(0.03)
Smith died on 26 February 1832 in [M] .	london	england(0.08), london(0.07), ireland(0.03)
Newton played as [M] during Super Bowl 50.	quarterback	quarterback(0.09), referee(0.05), mvp(0.05)
Warsaw is the most diverse [M] in Poland.	city	city(0.63), town(0.13), settlement(0.03)
Quran is a [M] text.	religious	religious(0.21), muslim(0.1), biblical(0.08)
president, and Thomas Watson, founder of [M].	ibm	ibm(0.21), microsoft(0.02), motorola(0.02)
Letham is a village in [M], Scotland.	angus	fife(0.52), angus(0.24), highland(0.06)
Hugh Ragin is an American [M] trumpeter.	jazz	jazz(0.97), classical(0.01), rock(0.01)
Avishkaar is a 1974 [M] movie.	hindi	bollywood(0.31), hindi(0.29), malayalam(0.11)
West of Bern, the population generally speaks [M].	french	german(0.72), french(0.13), italian(0.05)
He was succeeded as [M] by Christoph Ahlhaus.	mayor	chancellor(0.1), bishop(0.09), mayor(0.05)
His son Hugh became [M] of Saint-Gilles.	abbot	bishop(0.48), abbot(0.28), archbishop(0.13)
During his terms Romania joined [M].	nato	nato(0.25), yugoslavia(0.15), czechoslovakia(0.07)
It seized [M] and Czechoslovakia in 1938 and 1939.	austria	hungary(0.3), poland(0.23), austria(0.11)
Hostage Life was a Canadian punk band from [M].	toronto	toronto(0.25), vancouver(0.12), montreal(0.11)

Table 7: More factual probe examples of InformBERT on LAMA (Petroni et al., 2019). [M] denotes the masked token.

C GLUE Performance

We have conducted additional experiments on GLUE (Wang et al., 2018). InformBERT outperforms BERT-base on six out of nine tasks. Notably, InformBERT seems to underperform BERT by a large margin on CoLA, which is focused on the grammatical correctness. We suspect this is because InformBERT pays less attention to stop words that can be important for this task.

Model	GLUE (Wang et al., 2018)								
	CoLA	SST-2	MRPC	STS-B	QQP	MNLI	QNLI	RTE	WNLI
BERT-base	**56.53**	92.32	84.07	88.64	90.71	**83.91**	**90.66**	65.57	56.34
InformBERT	52.16	**92.66**	**87.50**	**88.75**	**90.90**	83.13	89.82	**65.70**	**56.93**

Table 8: Comparison of InformBERT and BERT-base on the dev. set of GLUE (Wang et al., 2018). both models are trained for 40 epochs using the same corpus. We report Matthews correlation for CoLA, Pearson correlation for STS-B and accuracy for other tasks.