Beyond Distributional Hypothesis: Let Language Models Learn Meaning-Text Correspondence

Myeongjun Jang[1] **Frank Mtumbuka**[1] **Thomas Lukasiewicz**[2,1]
[1] Department of Computer Science, University of Oxford, UK
[2] Institute of Logic and Computation, TU Wien, Austria
firstname.lastname@cs.ox.ac.uk

Abstract

The logical negation property (LNP), which implies generating different predictions for semantically opposite inputs (p is true iff $\neg p$ is false), is an important property that a trustworthy language model must satisfy. However, much recent evidence shows that large-size pre-trained language models (PLMs) do not satisfy this property. In this paper, we perform experiments using probing tasks to assess PLMs' LNP understanding. Unlike previous studies that only examined negation expressions, we expand the boundary of the investigation to lexical semantics. Through experiments, we observe that PLMs violate the LNP frequently. To alleviate the issue, we propose a novel intermediate training task, named *meaning-matching*, designed to directly learn a meaning-text correspondence, instead of relying on the distributional hypothesis. Through multiple experiments, we find that the task enables PLMs to learn lexical semantic information. Also, through fine-tuning experiments on 7 GLUE tasks, we confirm that it is a safe intermediate task that guarantees a similar or better performance of downstream tasks. Finally, we observe that our proposed approach[1] outperforms our previous counterparts despite its time and resource efficiency.

1 Introduction

Contemporary large-size PLMs, such as BERT (Devlin et al., 2019), ELECTRA (Clark et al., 2020), and GPT-2 and -3 (Radford et al., 2019; Brown et al., 2020), have shown excellent results in many downstream tasks, even performing better than humans in the GLUE (Wang et al., 2018) and SuperGLUE (Wang et al., 2019b) benchmark datasets.

However, their reliability is recently being challenged. Many studies have conducted various probing tasks and observed that PLMs exhibit faulty behaviours, such as insensitiveness to sentence ordering (Pham et al., 2021; Gupta et al., 2021; Sinha et al., 2021b), incomprehension on number-related representations (Wallace et al., 2019; Lin et al., 2020; Nogueira et al., 2021), and lack of semantic content understanding (Ravichander et al., 2020; Elazar et al., 2021). These issues raise concerns about PLMs' stability and reliability, precluding them from applications in practice, especially in risk-sensitive areas.

Another critical problem of PLMs is their inaccurate behaviour on *negation*, which is a principal property in many language understanding tasks. For tasks where the LNP holds (p is true iff $\neg p$ is false; see Aina et al. 2018), PLMs should make different answers for the original and negated inputs. However, several studies observed that PLMs violate this property. In masked knowledge retrieval tasks, PLMs frequently generate incorrect answers for negated input queries (Ettinger, 2020; Kassner and Schütze, 2020). In other studies, PLMs show a poor generalisation ability on negated natural language inference (NLI) datasets (Naik et al., 2018; Hossain et al., 2020).

Although the aforementioned studies produced promising analysis results, they limited the scope of the LNP only to adding negation expressions (e.g., "no" and "not"). However, other perturbations that generate the opposite meaning also can be applied to the property. Therefore, a consideration of such perturbation methods is necessary to fully assess whether PLMs satisfy the LNP.

Also, remedies to alleviate the problem have not been studied much yet. Hosseini et al. (2021) recently employed data augmentation and unlikelihood training (Welleck et al., 2020) to prevent models from generating unwanted words, given the augmented negated data during masked language modelling (MLM). However, this approach has several downsides. First, like previous works, Hosseini et al. (2021) only considered negation expressions. Second, the data augmentation method is contingent on many additional linguistic compo-

[1] https://github.com/MJ-Jang/beyond-distributional

nents, which causes the dependency of a model's performance on certain modules and precludes applying the method to other languages where such resources are unavailable. Finally, the model should be pre-trained from scratch with the unlikelihood objective, which consumes considerable time and resources.

In this paper, we expand the boundary of the LNP to lexical semantics, i.e., *synonyms* and *antonyms*, and ascertain that PLMs are prone to violate the LNP. Next, we propose a remedy, called **i**ntermediate-training on **m**eaning-**m**atching (IM^2), which hardly employs additional linguistic components. We hypothesise that a leading cause lies in the MLM training objective, which assumes the *distributional hypothesis* for learning the meaning of the text (Sinha et al., 2021a). Instead, we design a model that directly learns the correspondence between words and their semantic contents. Through experiments, we verify that our approach improves the model's comprehension of the LNP, while showing a stable performance on multiple downstream tasks.

Our main contributions are as follows: (i) We extend the investigation of the LNP from negation to lexical semantics (Section 2), (ii) we reveal that PLMs are prone to violate the LNP (Section 3), (iii) we propose a novel remedy, named IM^2, which is decoupled from the *distributional hypothesis* but learns meaning-text correspondence instead (Section 4), (iv) through experiments, we ascertain that the proposed approach improves the understanding of negation and lexical semantic information (Sections 5.1 and 5.2), and (v) we verify that meaning-matching is a stable and safe intermediate task that produces a similar or better performance in multiple downstream tasks (Sections 5.3 and 5.4).

2 Probing Tasks for Investigating the Logical Negation Property

We design three probing tasks to evaluate whether PLMs satisfy the LNP: masked knowledge retrieval on negated queries (MKR-NQ), masked word retrieval (MWR), and synonym/antonym recognition (SAR). Brief illustrations of each task are in Figure 1.

2.1 Masked Knowledge Retrieval on Negated Queries

The MKR-NQ task examines whether PLMs generate incorrect answers for negated queries. Following the work of Kassner and Schütze (2020), we constructed the evaluation dataset by negating the LAMA dataset (Petroni et al., 2019), which contains masked free-text forms of ConceptNet (Speer et al., 2017) triplets and their corresponding answers (e.g., (bird, CapableOf, fly) → ("A bird can [MASK]", fly)). The task aims to generate a correct word through MLM.

According to the LNP, a model must not generate the original answer if the query is negated. To measure how likely PLMs generate wrong predictions for negated queries, we collected pairs of (negated_query, wrong_predictions). We selected several relations in the LAMA dataset that ensure mutual exclusiveness between the original and negated queries.[2] For negating sentences, we selected LAMA data points that contain a single verb using the Spacy parts of speech (POS) tagger (Honnibal and Johnson, 2015). Next, we added negation expressions, such as "not" and "don't", or removed such expressions if they existed. Finally, we collected the wrong predictions from ConceptNet by using the head entity and relation. As a result, we collected 3,360 data points for this task. The list of the relations that we used and examples of the data are in Table 10 in Appendix A.

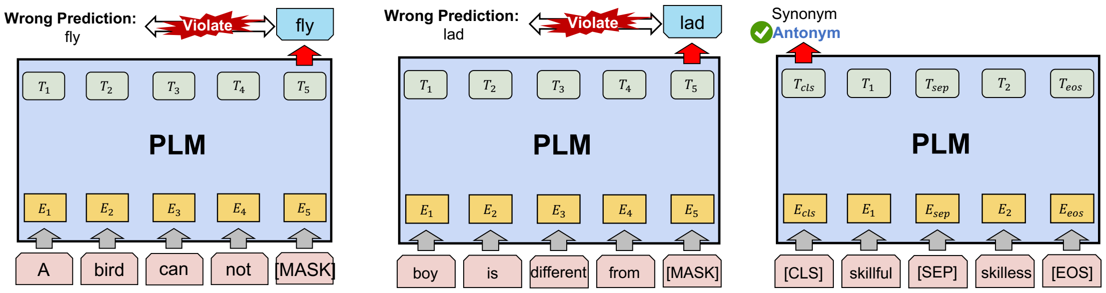
2.2 Masked Word Retrieval

To expand the boundary of the LNP to lexical semantics, we design the MWR task, which generates an answer of a masked query, asking for the synonym/antonym of a target word through MLM (e.g., "happy is the synonym of [MASK]").

Let s_w and a_w denote masked queries that ask the synonym and antonym of the word w, respectively. Also, let A_s and A_a refer to the list of correct answers for s_w and a_w, respectively. Intuitively, A_a becomes the wrong predictions of s_w, because s_w and a_w have the opposite meaning. Therefore, we can evaluate the violation of the LNP by investigating whether a PLM generates wrong predictions.

To extract commonly-used words for our experiment, we first extracted nouns, adjectives, and adverbs that appear more than five times in the SNLI dataset (Bowman et al., 2015). Among the extracted candidates, we filtered words that have synonyms or antonyms in ConceptNet. Finally, we generated masked queries by employing templates used by Camburu et al. (2020). As a result, we collected about 27K data points for MWR. The

[2] For example, the *HasProperty* relation is not suitable to use, because sentences like "Some adults are immature" and "Some adults are not immature" are not mutually exclusive.



(a) MKR-NQ (b) MWR (c) SAR

Figure 1: Illustration of the MKR-NQ, MWR, and SAR tasks.

templates and examples of the data are in Table 11 in Appendix A.

2.3 Synonym/Antonym Recognition

SAR is a classification that distinguishes whether two given words are synonyms or antonyms. It aims to evaluate whether the contextualised representations of PLMs reflect the lexical meaning of words. Therefore, we use a parametric probing model (Adi et al., 2017; Liu et al., 2019a; Belinkov and Glass, 2019; Sinha et al., 2021a) for the experiment. Specifically, the experiment is performed on the final layer of each PLMs, i.e., we only train the classifier while keeping the encoder frozen. We use ConceptNet to build the dataset. ConceptNet has much more synonym triplets compared to antonyms. As a result, we randomly sample the synonym triplets to maintain a balance. To that end, we collect 33K, 1K, and 2K data points for the train, dev, and test datasets, respectively.

2.4 Evaluation Metrics

We use the top-k hit rate (HR@k) to evaluate the performance on the MKR-NQ and MWR tasks. Assume that $P = \{(p_1, c_1), (p_2, c_2), \dots, (p_n, c_n)\}$ denotes the set of predictions for a data point x, where p_t and c_t refer to the predicted word and confidence score of the t-th prediction, respectively. Then, the top-k hit rate for a data point x is defined as follows:

$$HR@k(x) = \frac{\sum_{i=1}^{k} \mathbb{1}(p_i \in \mathcal{W}_x)}{k},$$

where $\mathcal{W}_x$ is the wrong prediction set of x. Intuitively, the metric measures the ratio of top-k predicted words that belong to the wrong prediction set.

To reflect the prediction confidence score to the evaluation metric, we additionally define the weighted top-k hit rate (WHR@k) that uses the confidence score as weights. It is worth to mention that lower metrics mean a better model performance in both cases as the metrics assess how likely the models make inaccurate answers that they must avoid. The weighted metric can be defined as follows:

$$WHR@k(x) = \frac{\sum_{i=1}^{k} c_i \times \mathbb{1}(p_i \in \mathcal{W}_x)}{\sum_{i=1}^{k} c_i}.$$

For the SAR task, we employ accuracy as an evaluation metric, because each data point has its own label, and the label distribution is not skewed.

3 PLMs Lack Information of Negation and Lexical Semantics

We select the following PLMs for the experiments: bidirectional encoder representations from transformers (BERT)-*base/large* (Devlin et al., 2019), RoBERTa-*base/large* (Liu et al., 2019b), and ALBERT-*base/large* (Lan et al., 2019). These PLMs are pre-trained with the MLM training objective. We added the ELECTRA-*small/base/large* models (Clark et al., 2020) for the SAR task, but it is not used for the MKR-NQ and MWR experiments, as the discriminator of the ELECTRA models are trained with the replaced token prediction (RTP) training objective and have no MLM classifier. No additional training is required for the MKR-NQ and MWR tasks. For the SAR task, we fine-tune each PLM for 10 epochs and apply the early stopping technique. We use the AdamW optimiser (Loshchilov and Hutter, 2019) for training with a learning rate of $5e^{-6}$ and a batch size of 32.

3.1 Results for MKR-NQ

The results for the MKR-NQ task are summarised in Table 1. In general, the results are consistent with previous works (Ettinger, 2020; Kassner and Schütze, 2020). We observe three important characteristics from the experimental results.

Model	MKR-NQ					MWR				
	HR@1	HR@3	WHR@3	HR@5	WHR@5	HR@1	HR@3	WHR@3	HR@5	WHR@5
BERT-*base*	9.57	6.38	8.42	5.00	7.81	35.03	18.83	28.71	13.26	26.03
BERT-*large*	13.33	7.70	11.17	6.03	10.51	36.66	20.45	29.68	14.60	26.56
RoBERTa-*base*	11.52	6.85	9.63	5.30	8.91	13.02	8.47	10.50	6.54	9.32
RoBERTa-*large*	15.72	9.25	13.31	6.86	12.24	27.92	17.07	23.06	12.84	20.82
ALBERT-*base*	4.24	3.75	4.24	3.26	4.09	26.37	14.95	22.10	10.75	20.32
ALBERT-*large*	9.22	6.38	7.96	4.94	7.30	50.77	25.05	42.67	17.03	39.09

Table 1: Overall results for the MKR-NQ and MWR experiments. We multiply 100 to each value to improve readability. Note that the lower the values the better.

	BERT		RoBERTa		ALBERT	
	base	*large*	*base*	*large*	*base*	*large*
$\mathcal{R}_{syn}$	37.79	36.58	17.13	46.01	11.37	76.10
$\mathcal{R}_{ant}$	41.44	41.79	13.57	30.21	33.26	62.65

Table 2: Ratios of instances that PLMs regenerate the word in the input sentence. $\mathcal{R}_{syn}$ and $\mathcal{R}_{ant}$ are the ratios of synonym and antonym-asking questions, respectively.

Model	Encoder-fixed		Fine-tune	
	$\mathcal{A}_{val}$	$\mathcal{A}_{test}$	$\mathcal{A}_{val}$	$\mathcal{A}_{test}$
BERT-*base* (108M)	53.1	55.0	84.0	85.6
BERT-*large* (333M)	54.4	53.5	92.1	92.5
RoBERTa-*base* (124M)	71.1	70.1	87.2	87.8
RoBERTa-*large* (355M)	69.7	69.1	93.7	94.2
ALBERT-*base* (11M)	56.6	58.1	81.5	84.0
ALBERT-*large* (17M)	54.7	56.6	86.9	88.0
ELECTRA-*small* (13M)	64.1	63.9	80.2	80.9
ELECTRA-*base* (109M)	67.9	70.6	93.3	92.9
ELECTRA-*large* (334M)	69.4	72.7	95.9	95.4

Table 3: Results of the SAR experiment. $\mathcal{A}_{val}$ and $\mathcal{A}_{test}$ are the accuracy of the *validation* and *test* dataset, respectively. We record the average of five repetitions.

First, large models produce a higher hit rate than their corresponding base-size models in all three PLMs, recording an average of about 1.5 times higher values. This implies that large-size models are more likely to generate wrong predictions for negated queries, even though they perform better than small-size models in many benchmark tests. The results suggest that evaluating a model's performance solely based on the accuracy metric is unwise.

Second, the hit rate decreases as k increases, which implies that the majority of PLMs' top predictions (e.g., k=1 or k=2) are incorrect. Finally, the weighted hit rate is much higher than the vanilla hit rate, suggesting that PLMs generate wrong predictions with high confidence.

3.2 Results for MWR

The results of the MWR task are summarised in Table 1. The three characteristics found in the MKR-NQ task are also observed in the MWR task. Also, we found the following additional patterns.

PLMs lack knowledge of antonyms. In general, the hit rates are extremely high compared to the MKR-NQ task in all the PLMs. Analysing their predictions, we find that PLMs generate incorrect predictions primarily in antonym-asking queries. Specifically, the average HR@1 of the antonym-asking queries is 41.9%, while that of the synonym-asking queries is only 1.4%. A leading cause is that PLMs simply replicate the word presented in the input query. Table 2 shows the ratio of instances where each PLM reproduces the same word in a question. While the values are quite high for both synonym-asking and antonym-asking queries, the problem is more severe in the latter case, because the generated predictions are definitely incorrect. Based on our results, we conclude that PLMs' contextualised representations lack lexical semantic information. Our conclusion is in line with the findings of Liu et al. (2019a) showing that encoder-fixed PLMs are not suitable to deal with tasks that require fine-grained linguistic knowledge.

Issues are more severe with nouns. We observe that the hit rates are higher when a word in a question is a noun. Specifically, the average HR@1 values of nouns, adjectives, and adverbs are 35.1%, 27.4%, and 11.8%, respectively. Interestingly, PLMs have a high error rate when dealing with nouns even though they are trained with a large written English corpus, where nouns form the greatest portion (at least 37%) of all POS tags (Hudson, 1994; Liang and Liu, 2013).

3.3 Results for SAR

As part of the comparison, we fine-tune each PLM on the SAR task, i.e., train the entire set of parameters. The results are summarised in Table 3. We observe a huge gap between the performance of fine-tuned models and that of encoder-fixed models.

In contrast to the fine-tuned models that produce a high accuracy, encoder-fixed models fall short of expectations, even recording almost a random guess performance in BERT models. Also, just as a common belief, large models' performance is greatly improved when fine-tuned. However, the difference between the large and small encoder-fixed models is insignificant, except for the ELECTRA models that exhibit only a marginal improvement. The two phenomenons suggest that PLMs' outstanding performance is predicated on updating many parameters to learn syntactic associations presented in training data (Niven and Kao, 2019; McCoy et al., 2019), but their contextualised representations do not carry abundant lexical meaning information.

4 Intermediate Training on Meaning Matching Task: IM^2

4.1 Issue of PLMs

Through the previous experiments, we observe that PLMs contain little information about negation and especially lexical semantics. We hypothesise a leading cause lies in the training objective of PLMs: the language modelling (LM) objective, which is a backbone pre-training task of almost all PLMs.

In the LM objective, words are generated based on given contexts. The *distributional hypothesis* (Harris, 1954), which assumes that semantically related or similar words will appear in similar contexts (Mrkšić et al., 2016), is the underpinning assumption of the LM objective (Sinha et al., 2021a). Under this assumption, a model learns the meaning of texts based on their correlation to others. This is a great benefit, because a model can learn the meaning of texts using only the text form, allowing unsupervised training. Based on this advantage, many unsupervised representations, such as Word2Vec (Mikolov et al., 2013), Glove (Pennington et al., 2014), and current PLMs, have been developed.

However, the problem is that the *distributional hypothesis* has limitations in reflecting a word's *semantic* meanings, because words having different or even opposite semantic meanings can appear in similar or the same contexts. For instance, consider the two words "boy" and "girl". We can readily imagine sentences in which the two words appear in the same context, e.g., "the little boy/girl cuddled the teddy bear closely". As a result, a model can learn their common functional meanings, i.e., young human beings, and the vector representations would be very similar if they were trained based on the *distributional hypothesis*. However, the representation hardly captures their semantic antonomy, e.g., gender. Similarly, negated sentences have almost identical contexts to their original forms. As a result, models cannot effectively learn the semantic meaning of words and negation expressions, provided they leverage only the text forms.

4.2 Meaning-Matching Task

In the light of meaning-text theory, there is a correspondence between linguistic expressions (*text*) and semantic contents (*meaning*) (Mel'čuk and Žolkovskij, 1970; Milićević, 2006). Instead of solely relying on the *distributional hypothesis*, we propose the new *meaning-matching* task, which can directly learn the correspondence. Specifically, meaning-matching is a classification that takes a word and a sentence as input and determines whether the sentence defines the word correctly. Through this task, a model can learn both meaning-text correspondences and correlations between a word and other words in a definition, which is rarely found in general corpora.

For training PLMs on our new task, we apply the *intermediate-training* technique (Phang et al., 2018; Wang et al., 2019a; Liu et al., 2019a; Pruksachatkun et al., 2020; Vu et al., 2020), which first fine-tunes PLMs on an intermediate task, and then fine-tunes the model again on target tasks. It has been shown that training on intermediate tasks that require high-level linguistic knowledge and inference ability could improve performance (Liu et al., 2019a; Pruksachatkun et al., 2020). Furthermore, it is more efficient in time and resources than pre-training models on large corpora (e.g., BERTNOT model (Hosseini et al., 2021)).

Dataset. We collect about 150K free-text definitions that depict the meaning of English words from **WordNet** (Miller, 1995) and the **English Word, Meaning, and Usage Examples** dataset.[3] In cases when a word appears in both datasets, we concatenate the word's definitions. Several examples of our data are presented in Table 12 in Appendix A. We use publicly available English datasets for convenience, but our approach is easily adaptable to other languages, since most of them have their own dictionaries.

[3] https://data.world/idrismunir/english-word-meaning-and-usage-examples/

Model	MKR-NQ					MWR				
	HR@1	HR@3	WHR@3	HR@5	WHR@5	HR@1	HR@3	WHR@3	HR@5	WHR@5
BERT-*large*	13.33	7.70	11.17	6.03	10.51	36.66	20.45	29.68	14.60	26.56
BERT-*large* (IM^2)	**11.41**	**7.01**	**9.86**	**5.57**	**9.14**	**18.92**	**13.07**	**15.78**	**10.30**	**14.14**
RoBERTa-*large*	15.72	9.25	13.31	6.86	12.24	27.92	17.07	23.06	12.84	20.82
RoBERTa-*large* (IM^2)	**6.56**	**4.97**	**6.05**	**3.99**	**5.67**	**22.08**	**12.68**	**18.94**	**9.20**	**17.63**

Table 4: Results of BERT-*large* and RoBERTa-*large* after applying the IM^2 approach. We multiply 100 to each value for a better readability. Note that the lower the values the better.

Model	Encoder-fixed		Fine-tune	
	$\Delta\mathcal{A}_{val}$	$\Delta\mathcal{A}_{test}$	$\Delta\mathcal{A}_{val}$	$\Delta\mathcal{A}_{test}$
BERT-*base* (108M)	+5.5*	+5.1*	+3.9*	+3.0*
BERT-*large* (333M)	+3.1*	+6.3*	+1.0	+0.2
RoBERTa-*base* (124M)	+4.5*	+5.9*	+1.3*	+1.6*
RoBERTa-*large* (355M)	+15.0*	+17.1*	+0.6	+0.5
ALBERT-*base* (11M)	-2.6	+2.5	+4.7*	+3.3*
ALBERT-*large* (17M)	+1.3	+1.4	+1.2	+1.6
ELECTRA-*small* (13M)	-4.1*	-2.7*	+1.1	+1.1
ELECTRA-*base* (109M)	+3.8*	+3.2*	-0.2	+0.7
ELECTRA-*large* (334M)	+14.0*	+10.2*	+0.4	+0.5

Table 5: PLMs' accuracy change in the SAR task when we apply IM^2. We record the average across 5 runs. Our models show a statistically significant difference with p-value $<$ 0.05 (*) compared to the baseline results in Table 3.

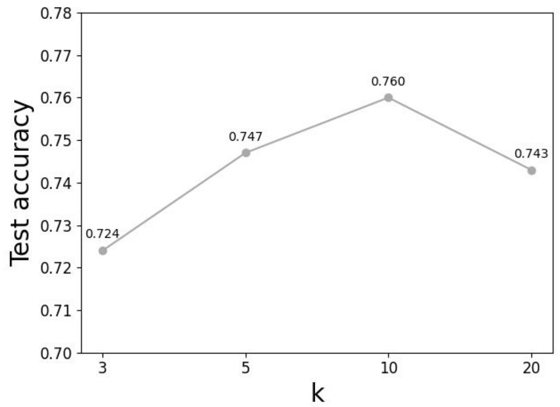


Figure 2: The performance of the RoBERTa-*base* (IM^2) model with different k values. We repeat each experiments for five times and record their average.

Training details. It is necessary to generate false word-definition pairs to train PLMs on the meaning-matching task. To achieve this, we use a negative sampling technique. We investigate the proper k in the range of 3, 5, 10, and 20. For a hyperparameter search, the performance of the RoBERTa-*base* model on the SAR task is used as a criterion. Figure 2 illustrates the SAR performance of the RoBERTa-*base* model with different k values. Intuitively, a large k value will lead the model to a better performance by investigating more word-meaning combinations. However, we observe that the model performs the best when k is 10, and the performance decreases if k is too large. We conjecture that a leading cause is that the dataset contains many words with similar meanings, mostly derived from the same *stem*. As a result, large k values can increase the possibility of recognising the meaning of such similar words as different.

QUERY: demand is an antonym of [MASK]	
RoBERTa-large demand	RoBERTa-large (IM^2) supply
QUERY: tomorrow is the opposite of [MASK]	
BERT-large tomorrow	BERT-large (IM^2) today
QUERY: question is an antonym of [MASK]	
BERT-large question	BERT-large (IM^2) answer

Table 6: Examples of top-1 predictions on MWR queries. Unlike the original PLMs, our models do not reproduce a word in a query and make quite accurate predictions.

To avoid the class-imbalance issue in a batch, we duplicate the correct word-definition pairs k times when we construct the training data. For training, the AdamW optimiser is used with a learning rate of $5e^{-6}$. We use 5% of data points for validation and train the models for 15 epochs with a batch size of 32. The early stopping technique is used to prevent overfitting.

5 Experiments and Results

We conduct the same probing tasks after the intermediate training on the meaning-matching task.[4]

5.1 SAR Results

We first focus on the SAR task. After the intermediate training, all models are fine-tuned on the SAR task with the same hyperparameters described in Section 3. The results are summarised in Table 5.

[4] Our models trained with the meaning-match task can be downloaded from the following repositories: ELECTRA-large, BERT-large, RoBERTa-large.

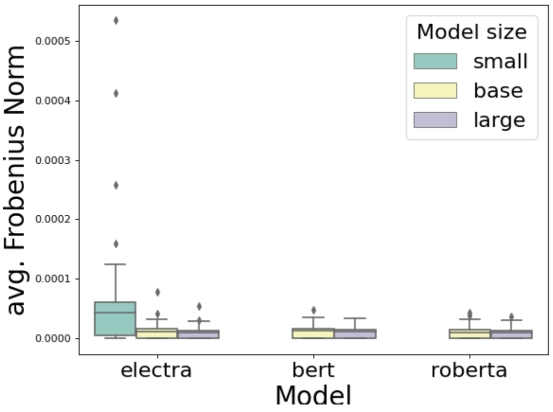


Figure 3: Frobenius norm box plots of PLMs' layer after intermediate training on the meaning-matching task.

Improved lexical semantic information. We generally observe marginal or no significant improvements when fine-tuning the whole parameters, especially for large-size PLMs. However, with fixed encoder, the performance is significantly improved for PLMs with more than 100M parameters, and the improvements are more significant for large PLMs. Our results show that the proposed approach assists PLMs to learn enhanced representations with more abundant lexical semantic information.

Catastrophic forgetting. We find that small PLMs, such as ELECTRA-*small* and ALBERT models, show no significant increase in performance or are negatively impacted. Because all PLMs achieve a comparable performance on the meaning-matching task, we hypothesise that a leading cause is *catastrophic forgetting* (Pruksachatkun et al., 2020; Wallat et al., 2020), where the model forgets previous knowledge learned through pre-training to accept new information from the intermediate task. To verify this, we measure the change of parameter values after IM^2. Concretely, let M_i and M_i^{mm} denote the parameter of i-th layer before and after IM^2. We calculate the average Frobenius norm for each layer:

$$\mathcal{F}_i = \frac{1}{|M_i|}|M_i - M_i^{mm}|_F.$$

Figure 3 shows the boxplots of $\mathcal{F}_i$ for each PLMs. We observe that the parameters of the ELECTRA-*small* model, which is negatively impacted, are changed considerably compared to other PLMs having parameters more than 100M. The results suggest that the size of PLMs is an important property to prevent the catastrophic forgetting issue.

5.2 MKR-NQ and MWR Results

Next, we perform the MKR-NQ and MWR tasks after applying the IM^2 method. Since our models are not trained with the MLM objective, we replace the encoder of original PLMs with that of the models after fine-tuning on the meaning-matching task and reuse the MLM classifier. For the experiments, we use BERT-*large* and RoBERTa-*large*, because they are pre-trained based on the MLM objective, and parameters are hardly changed after applying the IM^2 method. The results are summarised in Table 4.

We observe substantial decreases in the hit rates of incorrect predictions in both PLMs. For the MWR task, we find that the issue of regenerating a word in a given query is greatly relieved after applying the IM^2 method. Specifically, the percentage of such instances drops from 40.3% to 19.6% and from 33.8% to 25.2% for BERT-*large* and RoBERTa-*large*, respectively. Several examples of the predicted results are presented in Table 6. The results lend support to our claim that the IM^2 approach is of benefit to learning lexical semantic information and the meaning of negated expressions.

5.3 Fine-Tuning on the GLUE Benchmark

A critical drawback of intermediate training is that the target task performance could be negatively impacted if the intermediate task is not related to the target task (Liu et al., 2019a; Pruksachatkun et al., 2020). To confirm whether the issue occurs, we compare the performance of BERT, RoBERTa, and ELECTRA-*large* on 7 GLUE benchmark datasets (Wang et al., 2018) with their IM^2 counterparts. We train the models for 10 epochs for each dataset and apply the early stopping technique where the patience number is set to 3. It is observed that the training is generally finished within 8 epochs for all the models. The batch size per GPU and learning rates used for each dataset are described in Table 8. Datasets with large training set (e.g., MNLI, QNLI, and QQP) were not sensitive to the hyperparameters.

The results are presented in Table 7. We find no significant difference in performance for tasks with large datasets, such as MNLI, QNLI, QQP, and SST2. On the contrary, tasks with small datasets, like MRPC and RTE, are slightly improved. The

Model	COLA	MNLI-m	MNLI-mm	QNLI	RTE	QQP	MRPC	SST2
BERT-*large*	59.6±1.1	85.5±0.4	85.3±0.5	**91.7**±0.1	65.5±2.6	89.9±0.2	80.9±2.0	92.3±0.3
BERT-*large* (IM^2)	**61.5**±1.0	**85.7**±0.1	**85.5**±0.1	91.6±0.2	**66.8**±1.0	**90.0**±0.1	**82.8**±1.1	**92.4**±0.3
RoBERTa-*large*	62.9±1.9	90.2±0.1	**90.0**±0.2	**94.5**±0.1	81.7±1.8	90.9±0.4	87.2±1.1	**95.7**±0.1
RoBERTa-*large* (IM^2)	**64.8**±2.1	**90.3**±0.1	89.9±0.1	94.4±0.1	**83.1**±1.4	**91.0**±0.0	**88.2**±1.5	95.4±0.3
ELECTRA-*large*	68.4±2.3	**90.9**±0.1	90.7±0.2	**94.5**±0.3	86.9±2.2	91.6±0.5	88.9±1.5	**96.7**±0.1
ELECTRA-*large* (IM^2)	**69.1**±0.7	90.8±0.1	**90.7**±0.1	94.3±0.2	**87.0**±1.3	**91.7**±0.3	**89.5**±0.5	96.4±0.4

Table 7: GLUE benchmark validation performance of PLMs before and after intermediate training on the meaning-matching task. Matthew's correlation for the COLA and accuracy for the other tasks are used as an evaluation metric. We report the mean and standard deviation across 5 runs. The best values for each PLM are in bold.

	COLA	MNLI	QNLI	RTE	QQP	MRPC	SST2
b-size	16	64	64	8	64	8	64
lr	$2e^{-5}$	$1e^{-5}$	$1e^{-5}$	$2e^{-5}$	$1e^{-5}$	$1e^{-5}$	$1e^{-5}$

Table 8: Batch size and learning rates used for the GLUE benchmark experiments.

Model	SNLI		MNLI	
	dev	w/neg	dev	w/neg
BERTNOT	89.0±0.1	46.0±0.4	**84.3**±2.3	60.9±0.3
BERT-IM^2	**90.3**±0.2	**48.00**±0.5	83.1±0.3	**61.8**±0.6

Table 9: Accuracies on the original development dataset (dev) and the NegNLI (w/neg) dataset for SNLI and MNLI tasks. The results of our approach are averaged across 5 runs. The best values are in bold.

result is consistent with Pruksachatkun et al. (2020) and Vu et al. (2020), which showed that smaller tasks benefit much more from the intermediate training. Furthermore, unlike the previous studies that observed a negative transfer with the COLA dataset (Phang et al., 2018; Pruksachatkun et al., 2020), the performance is improved in our approach. The result suggests that meaning-matching is a safe intermediate task that ensures a positive transfer with target downstream tasks.

5.4 Experiments on the NegNLI Dataset

Finally, we conduct experiments on the NegNLI benchmark dataset (Hossain et al., 2020), where negation plays an important role for NLI tasks. As a baseline, we compare the reported performance of BERTNOT (Hosseini et al., 2021), which is a recently proposed remedy to improve PLMs' ability to understand negation. Since Hosseini et al. (2021) used BERT-*base* as a backbone model, we also apply the IM^2 method to BERT-*base*. The results are summarised in Table 9.

For both SNLI and MNLI, we observe that our approach outperforms BERTNOT in the NegNLI datasets, while yielding a comparable performance in the original development datasets. It is interesting that our approach improves the understanding of negation in both MKR-NQ and NegNLI tasks. We conjecture that a leading cause is that the definitions of the meaning-matching dataset contain many negation expressions, which enables a model to learn their proposed meaning (see Table 12). The results suggest that our proposed approach is more efficient than BERTNOT, because the IM^2 method leverages less time and resources for training.

6 Related Work

PLMs are at the core of many success stories in natural language processing (NLP). However, it remains unclear to what extent PLMs understand the syntactic and semantic properties of the human language. A series of probing tasks have been conducted on PLMs and have found them lacking or falling short on some language properties. Among the many findings of these probing tasks, PLMs have been found to be insensitive to the order of sentences when generating representations (Pham et al., 2021; Gupta et al., 2021; Sinha et al., 2021a), struggle to comprehend number-related representations (Wallace et al., 2019; Lin et al., 2020; Nogueira et al., 2021), and display a lack of semantic content understanding (Ravichander et al., 2020; Elazar et al., 2021).

In addition to the above faulty behaviours, Ettinger (2020) and Kassner and Schütze (2020) show that PLMs fail to comprehend *negation*, which is an important property of language in many natural language understanding (NLU) tasks. Ettinger (2020) check the ability of PLMs to understand the meaning of negation in given contexts. In their work, they check whether models are sensitive in their completions of sentences that either include *negation* or not. Under normal circumstances, the completions are expected to vary in truth depending on the presence or absence of negation in given sentences. Their results show that PLMs are insensitive to the impacts of negations when completing sentences. Kassner and Schütze (2020) construct the negated LAMA dataset by inserting negation elements (e.g., “not”) in the LAMA cloze questions (Petroni et al., 2019). They use negated and original question pairs to query PLMs and establish that models are equally prone to make the same predictions for both the original and negated questions. In a well-informed setting, it is expected that PLMs should make different predictions for the original and negated questions. This shows that PLMs struggle to comprehend negation.

In light of the highlighted faulty behaviours of PLMs, especially their struggle to comprehend negation, Hosseini et al. (2021) propose a remedy to alleviate the problem. In their remedy, they augment the language modelling objective with an unlikelihood objective (Welleck et al., 2020) based on negated sentences from the training corpus. They use a syntactic augmentation method to generate negated sentences. In this method, the dependency parse of the sentences, POS tags, and morphological information of each word are taken as input, and the negation of sentences is done using sets of dependency tree regular expression patterns, such as Semgrex (Chambers et al., 2007). During training, they replace objects in negated sentences with [MASK] tokens and use unlikelihood training to make the masked-out tokens unlikely under the PLM distribution. To ensure that negated sentences are factually false, they use the corresponding positive sentences as context for the unlikelihood prediction task.

Previous studies (e.g., Kassner and Schütze (2020)) have mostly limited the scope of the logical negation property only to the negation expressions (e.g., “no” and “not”). However, the core spirit of this property is the *opposite meaning*, which is not only limited to the *negation*. Welleck et al. (2020) consider negating sentences using dependency tree regular expression patterns. This widens the scope of negation, as it is not only limited to the negation expressions “no” and “not”. However, their approach relies on other components, such as Semgrex, and dependency and POS parsers, which could impact the quality of the data, hence impact the models’ performance. In this work, we consider other perturbation methods to generate the opposite-meaning sentences to investigate whether PLMs satisfy the logical negation property, and we propose a remedy, called **i**ntermediate-training on **m**eaning-**m**atching (IM^2), which hardly employs additional linguistic components.

7 Summary and Outlook

In this work, we investigated PLMs’ LNP. Compared to previous works that only examine negation expressions, we expanded the boundary of LNP to lexical semantics. We confirmed that PLMs are likely to violate LNP through extensive experiments.

We hypothesise that the distributional hypothesis is an insufficient basis for understanding the semantic meaning of texts. To alleviate the issue, we proposed a novel intermediate task: meaning-matching. Via experiments, we verified that meaning-matching is a stable intermediate task that substantially improves PLMs’ understanding of negation and lexical semantic information while guaranteeing a positive transfer with multiple downstream tasks. Also, our approach produces a better performance on the negated NLI datasets compared to the unlikelihood training-based method, which leverages much more time and resources. Our work suggests that it is time to move beyond the distributional hypothesis to develop logically consistent and stable language models.

Acknowledgements

This work was partially supported by the Alan Turing Institute under the EPSRC grant EP/N510129/1, by the AXA Research Fund, by the EPSRC grant EP/R013667/1, and by the ESRC grant “Unlocking the Potential of AI for English Law”. We also acknowledge the use of Oxford’s ARC facility, of the EPSRC-funded Tier 2 facilities JADE (EP/P020275/1) and JADE II (EP/T022205/1), and of GPU computing support by Scan Computers International Ltd. Frank Mtumbuka was supported by the Rhodes Trust under a Rhodes Scholarship.

References

Yossi Adi, Einat Kermany, Yonatan Belinkov, Ofer Lavi, and Yoav Goldberg. 2017. Fine-grained analysis of sentence embeddings using auxiliary prediction tasks. *International Conference on Learning Representations*.

Laura Aina, Raffaella Bernardi, and Raquel Fernández. 2018. A distributional study of negated adjectives and antonyms. In *Proceedings of the 5th Italian Conference on Computational Linguistics (CLiC-it 2018)*, volume 2253 of *CEUR Workshop Proceedings*.

Yonatan Belinkov and James Glass. 2019. Analysis methods in neural language processing: A survey. *Transactions of the Association for Computational Linguistics*, 7:49–72.

Samuel R. Bowman, Gabor Angeli, Christopher Potts, and Christopher D. Manning. 2015. A large annotated corpus for learning natural language inference. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (EMNLP 2015)*, pages 632–642.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc.

Oana-Maria Camburu, Brendan Shillingford, Pasquale Minervini, Thomas Lukasiewicz, and Phil Blunsom. 2020. Make up your mind! Adversarial generation of inconsistent natural language explanations. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4157–4165.

Nathanael Chambers, Daniel Cer, Trond Grenager, David Hall, Chloe Kiddon, Bill MacCartney, Marie-Catherine de Marneffe, Daniel Ramage, Eric Yeh, and Christopher D. Manning. 2007. Learning alignments and leveraging natural logic. In *Proceedings of the ACL-PASCAL Workshop on Textual Entailment and Paraphrasing*, pages 165–170.

Kevin Clark, Minh-Thang Luong, Quoc V. Le, and Christopher D. Manning. 2020. ELECTRA: Pre-training text encoders as discriminators rather than generators. *International Conference on Learning Representations*.

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, Volume 1 (Long and Short Papers)*, pages 4171–4186.

Yanai Elazar, Nora Kassner, Shauli Ravfogel, Abhilasha Ravichander, Eduard Hovy, Hinrich Schütze, and Yoav Goldberg. 2021. Measuring and improving consistency in pretrained language models. *Transactions of the Association for Computational Linguistics*, 9:1012–1031.

Allyson Ettinger. 2020. What BERT is not: Lessons from a new suite of psycholinguistic diagnostics for language models. *Transactions of the Association for Computational Linguistics*, 8:34–48.

Ashim Gupta, Giorgi Kvernadze, and Vivek Srikumar. 2021. Bert & family eat word salad: Experiments with text understanding. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 12946–12954.

Zellig S. Harris. 1954. Distributional structure. *Word*, 10(2-3):146–162.

Matthew Honnibal and Mark Johnson. 2015. An improved non-monotonic transition system for dependency parsing. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (EMNLP 2015)*, pages 1373–1378.

Md Mosharaf Hossain, Venelin Kovatchev, Pranoy Dutta, Tiffany Kao, Elizabeth Wei, and Eduardo Blanco. 2020. An analysis of natural language inference benchmarks through the lens of negation. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP 2020)*, pages 9106–9118.

Arian Hosseini, Siva Reddy, Dzmitry Bahdanau, R. Devon Hjelm, Alessandro Sordoni, and Aaron Courville. 2021. Understanding by understanding not: Modeling negation in language models. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1301–1312.

Richard Hudson. 1994. About 37% of word-tokens are nouns. *Language*, 70(2):331–339.

Nora Kassner and Hinrich Schütze. 2020. Negated and misprimed probes for pretrained language models: Birds can talk, but cannot fly. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7811–7818.

Zhenzhong Lan, Mingda Chen, Sebastian Goodman, Kevin Gimpel, Piyush Sharma, and Radu Soricut. 2019. ALBERT: A lite BERT for self-supervised learning of language representations. In *International Conference on Learning Representations*.

Junying Liang and Haitao Liu. 2013. Noun distribution in natural languages. *Poznań Studies in Contemporary Linguistics*, 49(4):509–529.

Bill Yuchen Lin, Seyeon Lee, Rahul Khanna, and Xiang Ren. 2020. Birds have four legs?! NumerSense: Probing numerical commonsense knowledge of pretrained language models. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP 2020)*, pages 6862–6868.

Nelson F. Liu, Matt Gardner, Yonatan Belinkov, Matthew E. Peters, and Noah A. Smith. 2019a. Linguistic knowledge and transferability of contextual representations. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 1073–1094.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019b. RoBERTa: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *International Conference on Learning Representations*.

Tom McCoy, Ellie Pavlick, and Tal Linzen. 2019. Right for the wrong reasons: Diagnosing syntactic heuristics in natural language inference. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3428–3448.

I. A. Mel'čuk and A. K. Žolkovskij. 1970. Towards a functioning 'meaning-text' model of language. *Linguistics*, 8(57):10–47.

Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S. Corrado, and Jeff Dean. 2013. Distributed representations of words and phrases and their compositionality. In *Advances in Neural Information Processing Systems*, volume 26. Curran Associates, Inc.

Jasmina Milićević. 2006. A short guide to the meaning-text linguistic theory. *Journal of Koralex*, 8:187–233.

George A. Miller. 1995. WordNet: A lexical database for English. *Communications of the ACM*, 38(11):39–41.

Nikola Mrkšić, Diarmuid Ó Séaghdha, Blaise Thomson, Milica Gašić, Lina M. Rojas-Barahona, Pei-Hao Su, David Vandyke, Tsung-Hsien Wen, and Steve Young. 2016. Counter-fitting word vectors to linguistic constraints. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 142–148.

Aakanksha Naik, Abhilasha Ravichander, Norman Sadeh, Carolyn Rose, and Graham Neubig. 2018. Stress test evaluation for natural language inference. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 2340–2353.

Timothy Niven and Hung-Yu Kao. 2019. Probing neural network comprehension of natural language arguments. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4658–4664.

Rodrigo Nogueira, Zhiying Jiang, and Jimmy Lin. 2021. Investigating the limitations of transformers with simple arithmetic tasks. *arXiv preprint arXiv:2102.13019*.

Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. GloVe: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP 2014)*, pages 1532–1543.

Fabio Petroni, Tim Rocktäschel, Sebastian Riedel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, and Alexander 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)*, pages 2463–2473.

Thang Pham, Trung Bui, Long Mai, and Anh Nguyen. 2021. Out of order: How important is the sequential order of words in a sentence in natural language understanding tasks? In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 1145–1160, Online. Association for Computational Linguistics.

Jason Phang, Thibault Févry, and Samuel R. Bowman. 2018. Sentence encoders on stilts: Supplementary training on intermediate labeled-data tasks. *CoRR*, abs/1811.01088.

Yada Pruksachatkun, Jason Phang, Haokun Liu, Phu Mon Htut, Xiaoyi Zhang, Richard Yuanzhe Pang, Clara Vania, Katharina Kann, and Samuel R. Bowman. 2020. Intermediate-task transfer learning with pretrained language models: When and why does it work? In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5231–5247.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

Abhilasha Ravichander, Eduard Hovy, Kaheer Suleman, Adam Trischler, and Jackie Chi Kit Cheung. 2020. On the systematicity of probing contextualized word representations: The case of hypernymy in BERT. In *Proceedings of the 9th Joint Conference on Lexical and Computational Semantics*, pages 88–102.

Koustuv Sinha, Robin Jia, Dieuwke Hupkes, Joelle Pineau, Adina Williams, and Douwe Kiela. 2021a. Masked language modeling and the distributional hypothesis: Order word matters pre-training for little. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing (EMNLP 2021)*, pages 2888–2913.

Koustuv Sinha, Prasanna Parthasarathi, Joelle Pineau, and Adina Williams. 2021b. UnNatural Language Inference. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 7329–7346.

Robyn Speer, Joshua Chin, and Catherine Havasi. 2017. ConceptNet 5.5: An open multilingual graph of general knowledge. In *Proceedings of the 31st AAAI Conference on Artificial Intelligence*.

Tu Vu, Tong Wang, Tsendsuren Munkhdalai, Alessandro Sordoni, Adam Trischler, Andrew Mattarella-Micke, Subhransu Maji, and Mohit Iyyer. 2020. Exploring and predicting transferability across NLP tasks. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP 2020)*, pages 7882–7926.

Eric Wallace, Yizhong Wang, Sujian Li, Sameer Singh, and Matt Gardner. 2019. Do NLP models know numbers? probing numeracy in embeddings. 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)*, pages 5307–5315.

Jonas Wallat, Jaspreet Singh, and Avishek Anand. 2020. BERTnesia: Investigating the capture and forgetting of knowledge in BERT. In *Proceedings of the Third BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP*, pages 174–183.

Alex Wang, Jan Hula, Patrick Xia, Raghavendra Pappagari, R. Thomas McCoy, Roma Patel, Najoung Kim, Ian Tenney, Yinghui Huang, Katherin Yu, Shuning Jin, Berlin Chen, Benjamin Van Durme, Edouard Grave, Ellie Pavlick, and Samuel R. Bowman. 2019a. Can you tell me how to get past Sesame Street? Sentence-level pretraining beyond language modeling. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4465–4476.

Alex Wang, Yada Pruksachatkun, Nikita Nangia, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. 2019b. SuperGLUE: A stickier benchmark for general-purpose language understanding systems. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. 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.

Sean Welleck, Ilia Kulikov, Stephen Roller, Emily Dinan, Kyunghyun Cho, and Jason Weston. 2020. Neural text generation with unlikelihood training. In *International Conference on Learning Representations*.

A Appendix: Examples

Relation	Negated Query	Wrong Predictions
IsA	Truth isn't a [MASK].	["fact", "statement", "concept", "actuality"]
CapableOf	A doctor cannot [MASK] you.	["care"]
PartOf	England isn't part of the [MASK].	["Europe"]
HasA	Apples don't have [MASK] inside them.	["stems", "seeds"]
UsedFor	A map isn't for [MASK].	["navigate", "locating", "navigating", "orienteering", "information"]
MadeOf	Air doesn't have [MASK].	["molecules"]
NotDesires	Soldier does want to be [MASK].	["die"]

Table 10: ConceptNet relations for constructing the MKR-NQ dataset and their corresponding sample data points.

Template	Query	Wrong Predictions
X is a synonym of Y	boy is a synonym of [MASK].	["sister", "girl"]
X is an antonym of Y	boy is an antonym of [MASK].	["boys", "brat", "man", "boy", "lad", . . .]
X is another form of Y	learning is another form of [MASK].	["forgetting", "teaching"]
X is the opposite of Y	learning is the opposite of [MASK].	["knowledge", "erudition", "eruditeness", "learning"]
X is a rephrasing of Y	speaker is a rephrasing of [MASK].	["microphone", "listener", "addressee"]
X is different from Y	speaker is different from [MASK].	["loudspeaker", "transducer", "talker", "speaker", . . .]

Table 11: Templates used to construct the MWR dataset and their sample data points.

Word	Definition
abnormal	not normal; not typical or usual or regular or conforming to a norm; out of ordinary; unusual
afebrile	having no fever
barefaced	with no effort to conceal
career	the particular occupation for which you are trained; a job or occupation that a person does for an extended period
cargo	goods carried by a large vehicle
revise	the act of rewriting something; to review, alter and amend, especially of written material
salary	something that remunerates; a determined yearly amount of money paid to an employee by an employer during a job

Table 12: Examples of word-definition pairs that we used for the meaning-matching task.