Double Trouble: How to not explain a text classifier’s decisions using counterfactuals synthesized by masked language models?

Anonymous ACL submission

Abstract

A principle behind dozens of attribution methods is to take the prediction difference between before-and-after an input feature (here, a token) is removed as its attribution—the individual treatment effect in causal inference. A recent popular Input Marginalization (IM) method (Kim et al., 2020) uses BERT to replace a token—i.e., simulating the do(·) operator—yielding more plausible counterfactuals. While Kim et al. (2020) reported that IM is effective, we find this conclusion not convincing as the Deletion\textsubscript{BERT} metric used in their paper is biased towards IM. Importantly, this bias should exist in many Deletion-based metrics, e.g., Insertion (Arras et al., 2017), Sufficiency, and Comprehensiveness (DeYoung et al., 2020). Furthermore, our rigorous evaluation using 6 metrics and on 3 datasets finds no evidence that IM is better than a Leave-One-Out (LOO) baseline. We provide two explanations for why IM is not better than LOO: (1) deleting a single word from the input only marginally reduces a classifier’s accuracy; and (2) a highly predictable word is always given near-zero attribution which may not match its true importance to the target classifier.

1 Introduction

Feature attribution maps (AMs), i.e., highlights indicating the importance of each input token w.r.t. a classifier’s decision, can help improve human accuracy on downstream tasks including detecting fake movie reviews (Lai and Tan, 2019) or identifying biases in text classifiers (Liu and Avci, 2019).

Many Leave-One-Out (LOO) methods compute the attribution of an input token by measuring the prediction change after substituting the token embedding with zeros (Li et al., 2016; Jin et al., 2020) or UNK (Kim et al., 2020). That is, deleting or replacing features is the underlying principle of at least 25 attribution methods (Covert et al., 2020). Based on evidence in computer vision (Bansal et al., 2020; Zhang et al., 2019), prior works in NLP hypothesized that removing a word from an input text forms out-of-distribution (OOD) inputs that yield erroneous AMs (Kim et al., 2020; Har...
becke and Alt, 2020) or AMs inconsistent with human’s perception of causality (Hase et al., 2021). To generate plausible counterfactuals, two teams of researchers (Kim et al., 2020; Harbecke and Alt, 2020) proposed Input Marginalization (IM), i.e. replace a word using BERT (Devlin et al., 2019) and compute an average prediction difference by marginalizing over all predicted words. Kim et al. (2020) claimed that IM yields more accurate AMs than the baselines that replace words by UNK or zeros but their quantitative results were reported only for one1 dataset and one evaluation metric.

In this paper, we re-assess their claim by, first, reproducing their IM results2, and then rigorously evaluating the effectiveness of IM on a diverse set of three datasets and six metrics. We found that:

- The DeletionBERT metric in Kim et al. (2020) is biased towards IM as both use BERT to replace words (Sec. 4). In contrast, the original Deletion metric (Arras et al., 2017) favors the LOOempty baseline as both delete words. This bias causes the incorrect interpretation that IM is better than LOO baselines in Kim et al. (2020) and exists in all Deletion variants including Insertion (Arras et al., 2017), Sufficiency, and Comprehensiveness (DeYoung et al., 2020).

- Under ROAR & ROARBERT (Hooker et al., 2019), the metrics that correct for the distribution shift in Deletion, LOOempty outperforms IM (Sec. 5.1). Compared to human annotations, LOOempty generates more plausible explanations than IM (Sec. 5.2). Under sanity check (Adebayo et al., 2018), IM is worse than LOOempty (Sec. 5.3). Overall, we find no evidence that IM is better than a simple LOOempty baseline on any of the above four metrics (which exclude the biased Deletion & DeletionBERT).

- To further test the main idea of IM—whether using BERT to generate plausible counterfactuals improves explainability—we integrate BERT into LIME (Ribeiro et al., 2016) but find that LIMEBERT only performs similarly to the original LIME (Sec. 6).

We argue that IM is not effective in practice because: (1) deleting a single word from an input has only a marginal effect to classification accuracy (Sec. 7.1); and (2) given a perfect, masked language model G, IM would be still unfaithful because highly predictable words according to G, e.g. “hot”, “air” in Fig.1, are always assigned near-zero attribution in IM regardless of how important they are to the classifier (Sec. 7.2). To our knowledge, our work is the first to question the commonly-assumed effectiveness of IM in NLP.

2 Methods and Related Work

Let \( f : \mathbb{R}^{n \times d} \rightarrow [0, 1] \) be a text classifier that maps a sequence \( x \) of \( n \) token embeddings, each of size \( d \), onto a confidence score of an output label. An attribution function \( A \) takes three inputs—a sequence \( x \), the model \( f \), and a set of hyperparameters \( H \)—and outputs a vector \( a = A(f, x, H) \in [-1, 1]^n \). Here, the explanation \( a \) associates each input token \( x_i \) to a scalar \( a_i \in [-1, 1] \), indicating how much \( x_i \) contributes for or against the target label.

Leave-One-Out (LOO) is a well-known method (Li et al., 2016) for estimating the attribution \( a_i \) by computing the confidence-score change after a token \( x_i \) is left out of the input \( x \), creating a shorter sequence \( x_{-i} \):

\[
a_i = f(x) - f(x_{-i}) \tag{1}
\]

This “prediction difference” (Robnik-Šikonja and Kononenko, 2008) is widely used as attribution in LOO methods (a.k.a “occlusion”) in both NLP (Jin et al., 2020) and Computer Vision (CV).

Under Pearl (2009) causal framework, the attribution \( a_i \) in Eq. 1 relies on a single, unrealistic counterfactual \( x_{-i} \) and thus is a biased estimate of the individual treatment effect (ITE):

\[
ITE = f(x) - \mathbb{E}[f(x) \mid do(T = 0)] \tag{2}
\]

where the binary treatment \( T \), here, is to keep or “realistically remove” the token \( x_i \) (i.e. \( T = 1 \) or 0) in the input \( x \), prior to the computation of \( f(x) \).

Removal techniques In CV, earlier attribution methods erase a feature by replacing it with (a) zeros (Zeiler and Fergus, 2014; Ribeiro et al., 2016); (b) random noise (Dabkowski and Gal, 2017; Lundberg and Lee, 2017); or (c) blurred versions of the original content (Fong et al., 2019). Yet, these hand-designed perturbation methods produce unrealistic counterfactuals that make AMs more unstable and less accurate (Bansal et al., 2020).
Recent works proposed to simulate the $do(T = 0)$ operator using an image inpainter. However, they either generate unnatural counterfactuals (Chang et al., 2019; Goyal et al., 2019) or only a single, plausible counterfactual per example (Agarwal and Nguyen, 2020).

**Input marginalization (IM)** In the parallel world of NLP, IM offers the closest estimate of the ITE. They compute the expectation term in Eq. 2 by marginalizing over many plausible textual counterfactuals generated by BERT:

$$E[f(x) \mid do(T = 0)] = \sum_{\tilde{x}_i \in V} p(\tilde{x}_i | x_{-i}) \cdot f(x_{-i}, \tilde{x}_i)$$ (3)

where $\tilde{x}_i$ is a token suggested by BERT (e.g., “hot”, “compressed”, or “open” in Fig. 1) with a likelihood of $p(\tilde{x}_i | x_{-i})$ to replace the masked token $x_i$. $V$ is the BERT vocabulary of 30,522 tokens. $f(x_{-i}, \tilde{x}_i)$ is the classification probability when token $x_i$ in the original input is replaced with $\tilde{x}_i$.

IM attribution is in the log space:

$$a_{IM} = \log-\text{odds}(f(x)) - \log-\text{odds}(E[f(x) \mid do(T = 0)])$$ (4)

where $\log-\text{odds}(p) = \log_2(p/(1 - p))$.

As computing the expectation term (Eq. 3) over the full vocabulary of size $\sim$30K is prohibitively slow, the authors only marginalized over the words that have a likelihood $\geq 10^{-5}$. We are able to reproduce the IM results of (Kim et al., 2020) by taking only the top-10 words and thus we use this setup for all experiments. Note that under BERT, the top-10 tokens, on average, already account for 81%, 90%, and 92% of the probability mass for SST-2, e-SNLI, and MultiRC, respectively.

**BERT** Like Kim et al. (2020), we use a pre-trained BERT “base”, uncased model (Devlin et al., 2019), from Huggingface (2020), to fill in a MASK token to synthesize plausible counterfactuals.

**LIME** We also test our findings of IM by integrating BERT into LIME, i.e. a more accurate attribution method (compared to LOO), which masks out multiple tokens at once to compute attribution.

LIME generates a set of randomly masked versions of the input, and the attribution of a token $x_i$, is effectively the mean classification probability over all the masked inputs when $x_i$ is not masked out. On average, each original LIME counterfactual has 50% of tokens taken out, often yielding text with large syntactic and grammatical errors.

**LIME$_{BERT}$** We use BERT to replace multiple masked tokens in each masked sentence generated by LIME to construct more plausible counterfactuals. However, for each word, we only use the top-1 highest-likelihood token given by BERT instead of marginalizing over multiple tokens because (1) the full marginalization is prohibitively slow; and (2) the top-1 token already carries most of the weight ($p \geq 0.81$; see Table 6).

### 3 Experiment framework

#### 3.1 Three datasets

We select a diverse set of three classification datasets that enable us to (1) compare with the results reported by Kim et al. (2020); and (2) assess AMs on six evaluation metrics (described in Sec. 3.3). These three tasks span from sentiment analysis (SST-2), natural language inference (e-SNLI) to question answering (MultiRC), covering a wide range of sequence length ($\sim$20, 24, and 299 tokens per example, respectively). SST-2 and e-SNLI were the two datasets where Kim et al. (2020) found IM to be superior to LOO baselines.

**SST** Stanford Sentiment Treebank (Socher et al., 2013) is a dataset of $\sim$12K RottenTomato movie-review sentences, which contain human-annotated sentiment annotations for phrases. Each phrase and sentence in SST is assigned a sentiment score in $[0, 1]$ ($0 = $ negative, $0.5 = $ neutral, $1 = $ positive).

**SST-2** has $\sim$70K SST examples (including both phrases and sentences) where the regression scores per example were binarized to form a binary classification task (Socher et al., 2013).

**e-SNLI** A 3-way classification task of detecting whether the relation between a premise and a hypothesis is entailment, neutral or contradiction (Bowman et al., 2015). e-SNLI has 569K instances of (input, label, explanation) where the explanations are crowd-sourced (Camburu et al., 2018).

**MultiRC** Multi-sentence Reading Comprehension (Khshabhi et al., 2018) is a multiple-choice question-answering task that provides multiple input sentences as well as a question and asks the

---

3We find replacing all tokens at once or one at a time to produce similar LIME$_{BERT}$ results.
model to select one or multiple correct answer sentences. MultiRC has ∼6K examples with human-annotated highlights at the sentence level.

### 3.2 Classifiers
Following Kim et al. (2020); Harbecke and Alt (2020); Hase et al. (2021), we test IM and LOO baselines in explaining BERT-based classifiers.

For each task, we train a classifier by fine-tuning the entire model, which consists of a classification layer on top of the pre-trained BERT (described in Sec. 2). The dev-set top-1 accuracy scores of our SST-2, e-SNLI, & MultiRC classifiers are 92.66%, 90.92%, and 69.10%, respectively. On the SST bisarined dev-set, which contains only sentences, the SST-2-trained classifier’s accuracy is 87.83%.

#### Hyperparameters
Following the training scheme of HuggingFace, we fine-tune all classifiers for 3 epochs using Adam optimizer (Kingma and Ba, 2015) with a learning rate of 0.00002, $\beta_1 = 0.9$, $\beta_2 = 0.999$, $\epsilon = 10^{-8}$. A batch size of 32 and a max sequence length of 128 are used for SST-2 and e-SNLI while these hyperparameters for MultiRC are 8 and 512, respectively. Dropout with a probability of 0.1 is applied to all layers. Each model was trained on an NVIDIA 1080Ti GPU.

### 3.3 Six evaluation metrics
As there are no groundtruth explanations in XAI, we use six common metrics to rigorously assess IM’s effectiveness. For each classifier, we evaluate the AMs generated for all dev-set examples.

#### Deletion
is similar to “Comprehensiveness” (DeYoung et al., 2020) and is based on the idea that deleting a token of higher importance from the input should cause a larger drop in the output confidence score. We take the original input and delete one token at a time until 20% of the tokens in the input is deleted. A more accurate explanation is expected to have a lower Area Under the output-probability Curve (AUC) (Arras et al., 2017).

#### Deletion$_{\text{BERT}}$ a.k.a. AUC$_{\text{rep}}$ in Kim et al. (2020), is the Deletion metric but where a given token is replaced by a BERT top-1 suggestion instead of an empty string. Deletion$_{\text{BERT}}$ was proposed to minimize the OOD-ness of samples introduced by deleting words in the original Deletion metric, i.e. akin to integrating BERT into LOO to create IM.

#### RemOve And Retrain (ROAR)
To avoid a potential OOD generalization issue caused by the Deletion metric, a common alternative is to retrain the classifier on these modified inputs (where N% of the highest-attribution words are deleted) and measure its accuracy drop (Hooker et al., 2019). A more faithful attribution method is supposed to lead to a re-trained classifier of lower accuracy as the more important words have been deleted from the training examples. For completeness, we also implement ROARSEBT, which uses BERT to replace the highest-attribution tokens instead of deleting them without replacement.

#### Agreement with human-annotated highlights
To assess the plausibility of AMs to serve as a text-highlighter, a common metric is to compare an AM with the tokens that humans deem indicative of the groundtruth label (Wiegreffe and Marasović, 2021).

Because human annotators only label the tokens supportive of a label (e.g. Fig. 2), when comparing AMs with human annotations, we zero out the negative values in AMs. Following Zhou et al. (2016), we binarize a resulting AM at an optimal threshold $\tau$ in order to compare it with human-annotated highlights under Precision@1.

#### Sanity check
(Adebayo et al., 2018) is a well-known metric for testing insensitivity (i.e. bias) of attribution methods w.r.t. model parameters. For ease of interpretation, we compute the % change of per-word attribution values in sign and magnitude as we randomize the classification layer’s weights. A better attribution method is expected to be more sensitive to weight randomization.

### 4 Bias of Deletion metric and its variants
In explaining SST-2 classifiers, we successfully reproduce the AUC$_{\text{rep}}$ results reported in Kim et al. (2020), i.e. IM outperformed LOO$_{\text{zero}}$ and LOO$_{\text{unk}}$, which were implemented by replacing a word with the PAD and UNK token of BERT, respectively (Table 1). However, we hypothesize that Deletion$_{\text{BERT}}$ is biased towards IM as both use BERT to replace words, yielding a false sense of IM effectiveness reported in Kim et al. (2020).

To test this hypothesis, we add another baseline of LOO$_{\text{empty}}$, which was not included in Kim et al. (2020), i.e. erasing a token from the input without replacement (Eq. 1), mirroring the original Deletion metric. To compare with IM, all LOO methods in this paper are also in the log-odds space.

#### Results
Interestingly, we find that, under Deletion,
on both SST-2 and e-SNLI, IM underperformed all three LOO baselines and that LOOempty is the highest-performing method (Table 1). In contrast, IM is the best method under DeletionBERT. To our knowledge, our work is the first to document this bias of the Deletion metric widely used in the literature (Hase et al., 2021; Wiegreffe and Marasović, 2021; Arras et al., 2017). This bias, in principle, should also exist in other Deletion variants including Insertion (Arras et al., 2017), Sufficiency, and Comprehensiveness (DeYoung et al., 2020).

<table>
<thead>
<tr>
<th>Task</th>
<th>Metrics ↓</th>
<th>IM</th>
<th>LOOempty</th>
<th>LOOunk</th>
<th>LOOempty</th>
</tr>
</thead>
<tbody>
<tr>
<td>SST-2</td>
<td>Deletion</td>
<td>0.4732</td>
<td>0.4374</td>
<td>0.4464</td>
<td>0.4241</td>
</tr>
<tr>
<td></td>
<td>Deletion</td>
<td>0.4922</td>
<td>0.4970</td>
<td>0.5047</td>
<td>0.5065</td>
</tr>
<tr>
<td>c-SNLI</td>
<td>Deletion</td>
<td>0.3912</td>
<td>0.2798</td>
<td>0.3742</td>
<td>0.2506</td>
</tr>
<tr>
<td></td>
<td>Deletion</td>
<td>0.2816</td>
<td>0.3240</td>
<td>0.3636</td>
<td>0.3328</td>
</tr>
</tbody>
</table>

Table 1: IM is the best method under DeletionBERT, as reported in Kim et al. (2020), but the worst under Deletion. Both metrics measure AUC (lower is better).

5 No evidence that IM is better than LOO

To avoid the critical bias of Deletion and DeletionBERT, we further compare IM and LOO on four common metrics that are not Deletion-based.

5.1 Under ROAR and ROARBERT, IM is on-par with or worse than LOOempty

A lower AUC under Deletion may be the artifact of the classifier misbehaving under the distribution shift when one or multiple input words are deleted. ROAR (Hooker et al., 2019) was designed to ameliorate this issue by re-training the classifier on the modified training-set (where the top N% highest-attribution tokens in each example are deleted) before evaluating their accuracy.

To more objectively assess IM, we use ROAR and ROARBERT metrics to compare IM vs. LOOempty (i.e. the best LOO variant in Table 1).

Experiment For both IM and LOOempty, we generate AMs for every example in the SST-2 train and dev sets, and remove N% highest-attribution tokens per example to create new train and dev sets. We train 5 models on the new training set and evaluate them on the new dev set. We repeat ROAR and ROARBERT with N ∈ {10, 20, 30}.5

Results As more tokens are removed (i.e. N increases), the mean accuracy of 5 models gradually decreases (Table 2; from 92.66% to ~67%). Under both ROAR and ROARBERT, the models trained on the new training set derived from LOOempty AMs often obtain lower mean accuracy than those of IM (Table 2a vs. b). At N = 10% under ROAR, LOOempty outperforms IM (Table 2; 74.59 vs. 76.22), which is statistically significant (2-sample t-test, p = 0.037). In all other cases, the difference between IM vs. LOOempty is not statistically significant.

In sum, under both ROAR and ROARBERT, IM is not more faithful than LOOempty.

5.2 LOOempty explanations are better aligned with human annotations than IM

To increase our understanding of the differences between LOOempty and IM, here, we compare them against the human highlights for SST, e-SNLI, and MultiRC. While human highlights are not exactly groundtruth explanations, this human-alignment metric is a common proxy for the correctness of AMs in both NLP (Wiegreffe and Marasović, 2021) and computer vision (Zhou et al., 2016).

Annotation preprocessing To control for quality, we preprocess the human annotations in each dataset as the following. In SST, where each sentence has multiple phrases labeled with a sentiment score ∈ [0, 1] (0.5 being the “neutral” midpoint), we only use the phrases that have high-confidence sentiment scores, i.e. ≤ 0.3 (for “negative”) or ≥ 0.7 (for “positive”). Also, we do not use the annotated phrases that are too long, i.e., longer than 50% of the sentence length.

Each token in an e-SNLI example are labeled “important” by between 0–3 annotators. To filter out noise, we only use the tokens that are highlighted by at least two or three annotators (hereafter “L2” and “L3” subsets, respectively).

A MultiRC example contains a question and a paragraph where each sentence is labeled “important” or “unimportant” to the groundtruth answer (Fig. A5). We convert these sentence-level highlights into token-level highlights to compare them with the binarized AMs of IM and LOOempty.

Experiment We run IM and LOOempty on the BERT-based classifiers on the dev set of SST, e-SNLI, and MultiRC. All AMs generated are binarized using a threshold τ ∈ {0.05x | 0 < x < 20 and x ∈ N}. We compute the average IoU, pre-
Table 2: Dev-set mean accuracy (%) of 5 models trained on the new SST-2 examples where $N\%$ of highest-attribution words per example are removed (i.e. ROAR) or replaced via BERT (i.e. ROAR\textsubscript{BERT}). On average, under both metrics, LOO\textsubscript{empty} (a) is a slightly better, i.e. lower mean accuracy, than IM (b). Notably, LOO\textsubscript{empty} statistically significantly outperforms IM under ROAR at $N = 10\%$ (2-sample t-test; $p = 0.037$) (d). Both LOO\textsubscript{empty} and IM substantially outperform a random baseline (c) that considers $N\%$ random tokens important.

Table 3: Compared to IM, LOO\textsubscript{empty} is substantially more consistent with human annotations over all three datasets. Note that the gap between LOO\textsubscript{empty} and IM is $\sim 3\times$ wider when comparing AMs with the e-SNLI tokens that at least three annotators label “important” (i.e. L3), compared to L2 (higher is better). LIME\textsubscript{BERT} explanations are slightly less consistent with human highlights than those of LIME (a) despite their counterfactuals are more realistic.

Figure 2: LOO\textsubscript{empty} binarized attribution maps align better with human highlights than those of IM.

**e-SNLI and MultiRC results** Similarly, in both tasks, LOO\textsubscript{empty} explanations are more consistent with human highlights than IM explanations under all four metrics (see Table 3b–d and qualitative examples in Figs. 3 & A8–A11).

Remarkably, in MultiRC where each example is substantially longer ($\sim$299 tokens per example) than those in the other tasks, the recall and F1 scores of LOO\textsubscript{empty} is, respectively, $2 \times$ and $4 \times$ higher than those of IM (see Table 3).

Figure 3: LOO\textsubscript{empty} important words are in a stronger agreement with human highlights (P) and hypothesis (H).
5.3 IM less sensitive to model randomization

Adebayo et al. (2018) found that many attribution methods can be surprisingly biased, i.e. insensitive to even randomization of the classifier’s parameters. Here, we test the degree of insensitivity of IM when the last classification layer of BERT-based classifiers is randomly re-initialized. We use three SST-2 classifiers and three e-SNLI classifiers.

Surprisingly, IM is consistently worse than LOO empty, i.e. more insensitive to classifier randomization. That is, on average, the IM attribution of a word changes signs (from positive to negative or vice versa) less frequently, e.g. 62.27% of the time, compared to 71.41% for LOO empty on SST-2 (Table 4a). The average change in attribution magnitude of IM is also ∼1.5× smaller than that of LOO empty (Table 4b). For example, the IM attribution of “hot”, “air”, “balloon”, and “racing” in Fig. 1 remain consistently unchanged near-zero even when the classifier is randomized three times because these words are highly predictable by BERT from the context (Fig. 1b; IM1 to IM3). More in-depth explanations in Sec. 8.

<table>
<thead>
<tr>
<th>Criteria</th>
<th>Method</th>
<th>SST-2</th>
<th>e-SNLI</th>
</tr>
</thead>
<tbody>
<tr>
<td>(a) % tokens changing sign</td>
<td>LOO empty</td>
<td>71.41±17.12</td>
<td>56.07±21.82</td>
</tr>
<tr>
<td></td>
<td>IM</td>
<td>62.27±17.75</td>
<td>49.57±20.35</td>
</tr>
<tr>
<td>(b) Average absolute differences</td>
<td>LOO empty</td>
<td>0.46±0.18</td>
<td>0.26±0.14</td>
</tr>
<tr>
<td></td>
<td>IM</td>
<td>0.31±0.12</td>
<td>0.16±0.12</td>
</tr>
</tbody>
</table>

Table 4: The percentage (%) of token (a) whose attribution scores change signs and (b) the average of absolute differences in attribution magnitude after classifier randomization (higher is better). IM is consistently more insensitive than LOO empty in both SST-2 and e-SNLI.

6 LIMEBERT attribution maps are not more aligned with human annotations

Sec. 5 shows that replacing a single word by BERT (instead of deleting) creates more realistic inputs but does not improve AM quality w.r.t. LOO.

Here, we test whether this conclusion generalizes to the case of LIME where multiple words are deleted from a sentence. Similar to Sec. 5.2, we compare LIME and LIMEBERT AMs with human SST annotations (but not the Deletion-derived metrics due to their bias described in Sec. 4).

**Experiment** We use the default hyperparameters of the original LIME (Ribeiro, 2021) for both LIME and LIMEBERT. The number of counterfactual samples was 1,000 per example.

**Results** Although LIMEBERT counterfactuals are more natural, the derived explanations are surprisingly less plausible to human than those generated by the original LIME. That is, compared to human annotations in SST, LIMEBERT’s IoU, precision and F1 scores are all slightly worse than those of LIME (Table 3a). Consistent with the IM vs. LOO comparison in Sec. 5.2, replacing one or more words (instead of deleting them) using BERT in LIME generates AMs that are similarly or less aligned with humans. In sum, for both LOO and LIME, we find no evidence that using realistic counterfactuals from BERT provides improvement in AM quality.

To minimize the possibility that the pre-trained BERT is suboptimal in predicting missing words on SST-2, we also finetune BERT using the mask-language modeling objective on SST-2 and repeat the experiment in this section. Yet, interestingly, we find the above conclusion to not change (Table 3a; LIMEBERT_SST is worse than LIME).

7 Why is IM not better than LOO empty?

Despite being ≥ 5× more computationally expensive\(^6\), IM is not better than LOO empty on all four unbiased tested metrics (Sec. 5). We provide two explanations for this surprising finding.

7.1 Classification accuracy only drops marginally when a token is removed

Prior works argue that feeding unnatural LOO samples into classifiers may yield more unfaithful AMs (Kim et al., 2020; Harbecke and Alt, 2020; Hase et al., 2021); however, our rigorous evaluation using 3 datasets and 6 evaluation metrics did not find any case where IM outperforms LOO (except under the biased DeletionBERT metric).

To understand why using more plausible samples did not improve AM explainability, we assess the Δ drop in classification accuracy when a word is deleted (i.e., LOO empty samples; Fig. A12) and the Δ when a word is replaced using BERT (i.e. IM samples).

**Results** Across SST, e-SNLI, and MultiRC, the accuracy scores of classifiers only drop marginally ~1-4% (Table 5) when a single token is deleted.

\(^6\)On a 1080Ti GPU, it took IM ∼15 mins to complete the full SST dev-set compared to ∼3 mins for LOO empty. These numbers are ∼105 hrs vs. ∼10.5 hrs on the MultiRC dev-set.
Our findings are aligned with IM’s low recall. That is, if BERT fills in an exact-match \( \tilde{x}_i \) for an original word \( x_i \), the prediction difference for this replacement \( \tilde{x}_i \) will be 0 in Eq. 4. Furthermore, a high likelihood of \(~0.81\) for \( \tilde{x}_i \) sets an empirical upper-bound of 0.19 for the attribution of the word \( x_i \), which explains the insensitivity of IM to classifier randomization (Fig. 1; IM\(_1\) to IM\(_3\)).

The analysis here is also consistent with our additional findings that IM attribution tends to be smaller than that of LOO\(\text{empty}\) and therefore leads to heatmaps of lower coverage of the words labeled “important” by humans (see Sec. A).

8 Discussion and Conclusion

Our series of rigorous experiments reveal severe shortcomings of IM in prior work (Kim et al., 2020; Harbecke and Alt, 2020) for explaining a classifier’s decisions in NLP at the word level. Consider an example: In the input text, there exists a token \( x_i \) that is \(~100\%\) predictable by BERT from the context, i.e. \( p(\tilde{x}_i|x_{\sim i}) \approx 1 \) (see hot, air or balloons in Fig. 1). Then, by applying Equations 3 & 4, the attribution \( a_i \approx 0 \), by construction, regardless of how important the token \( x_i \) is to the classifier.

A solution may be to leave such \( x_i \) token out of the marginalization (Eq. 3), i.e. only marginalizing over the other tokens suggested by BERT. However, these other replacement tokens altogether have a sum likelihood of 0. That is, replacing token \( x_i \) by zero-probability tokens (i.e. truly implausible) would effectively generate OOD text, which, in turn is not desired (Hase et al., 2021).

Another possible fix for IM in NLP might be to operate at the phrase level instead of word level as deleting a set of contiguous words has a larger effect to the classifier predictions. However, in general, using a black-box generative model in the process of explaining the decision of another black-box classifier may be “Double Trouble” in any domain (Rudin, 2019).
References


Appendix for:
**Double Trouble: How to not explain a text classifier’s decisions using counterfactuals synthesized by masked language models?**

A IM explanations have smaller attribution magnitude per token and lower word coverage

To further understand the impact of the fact that BERT tends to not change a to-remove token (Sec. 7.2), here, we quantify the magnitude of attribution given by IM and its coverage of important words in an example.

**Smaller attribution magnitude** Across three datasets, the average absolute values of attribution scores (which are $\in [-1, 1]$) of IM are not higher than that of LOO empty (Table A1). Especially in MultiRC, IM average attribution magnitude is $4.5 \times$ lower than that of LOO empty (0.02 vs 0.09).

**Lower word coverage** We define coverage as the average number of highlighted tokens per example (e.g. Fig. 1) after binarizing a heatmap at the method’s optimal threshold.

The coverage of LOO empty is much higher than that of IM on SST (40% vs 30%) and MultiRC examples (27% vs 6%), which is consistent with the higher recall of LOO empty (Table A2; a vs. b). For e-SNLI, although IM has higher coverage than LOO empty (14% vs. 10%), the coverage of LOO empty is closer to the human coverage (9%). That is, IM assigns high attribution incorrectly to many words, resulting in a substantially lower precision than LOO empty, according to e-SNLI L3 annotations (Table 3b; 0.3814 vs. 0.4687).

In sum, **chaining our results together**, we found BERT to often replace a token $x_i$ by an exact-match with a high likelihood (Sec. 7.2), which sets a low empirical upper-bound on attribution values of IM, causing IM explanations to have smaller attribution magnitude. As the result, after binarization, fewer tokens remain highlighted in IM binary maps (e.g. Fig. 3).

<table>
<thead>
<tr>
<th>Method</th>
<th>SST</th>
<th>e-SNLI</th>
<th>MultiRC</th>
</tr>
</thead>
<tbody>
<tr>
<td>LOO empty</td>
<td>0.22 ± 0.27</td>
<td>0.15 ± 0.24</td>
<td>0.09 ± 0.09</td>
</tr>
<tr>
<td>IM</td>
<td>0.17 ± 0.27</td>
<td>0.15 ± 0.27</td>
<td>0.02 ± 0.09</td>
</tr>
</tbody>
</table>

Table A1: The average absolute value of attribution scores per token of LOO empty is consistently higher than that of IM.
Table A2: Compared to IM, the coverage of LOO\textsubscript{empty} is closer to the coverage of human explanations.

<table>
<thead>
<tr>
<th>Explanations generated by</th>
<th>SST</th>
<th>e-SNLI</th>
<th>MultiRC</th>
</tr>
</thead>
<tbody>
<tr>
<td>(a) LOO\textsubscript{empty}</td>
<td>40%</td>
<td>19%</td>
<td>10%</td>
</tr>
<tr>
<td>(b) IM</td>
<td>30%</td>
<td>21%</td>
<td>14%</td>
</tr>
<tr>
<td>(c) Human</td>
<td>37%</td>
<td>18%</td>
<td>9%</td>
</tr>
</tbody>
</table>

# tokens per example 20 24 299

SST example. Groundtruth: “positive”

S: may not have generated many sparks, but with his affection for Astoria and its people he has given his tale a warm glow.

S:\textsubscript{t} may not have generated many sparks, but with his affection for Astoria and its people he has given his tale a warm glow.

Figure A1: BERT often correctly predicts the masked tokens (denoted in red, green, blue rectangles) and assigns a high likelihood to the tokens that are labeled important by humans in the SST “positive” example. In each panel, we show the top-3 tokens suggested by BERT and their associated likelihoods.

SST example. Groundtruth: “negative”

S: Villeneuve spends too much time wallowing in Bibi’s generic angst (there are a lot of shots of her gazing out windows).

S:\textsubscript{t} Villeneuve spends too much time wallowing in Bibi’s generic angst (there are a lot of shots of her gazing out windows).

Figure A2: BERT often correctly predicts the masked tokens (denoted in red, green, blue rectangles) and assigns a high likelihood to the tokens that are labeled important by humans in the SST “negative” example. In each panel, we show the top-3 tokens suggested by BERT and their associated likelihoods.
Figure A3: BERT often correctly predicts the masked tokens (denoted in red, green, blue rectangles) and assigns a high likelihood to the tokens that are labeled important by humans in the e-SNLI “entailment” example which contains a pair of premise (P) and hypothesis (H). In each panel, we show the top-3 tokens suggested by BERT and their associated likelihoods.

![Image of e-SNLI example]

Figure A4: BERT often correctly predicts the masked tokens (denoted in red, green, blue rectangles) and assigns a high likelihood to the tokens that are labeled important by humans in the e-SNLI “neutral” example which contains a pair of premise (P) and hypothesis (H). In each panel, we show the top-3 tokens suggested by BERT and their associated likelihoods.

![Image of e-SNLI example]

MultiRC example. Groundtruth & Prediction: “True” (confidence: 0.98)

Figure A5: BERT often correctly predicts the masked tokens (denoted in red, green, blue rectangles) and assigns a high likelihood to the tokens that are labeled important by humans in the MultiRC “True” example which contains a triplet of paragraph (P), question (Q) and answer (A). In each panel, we show the top-3 tokens suggested by BERT and their associated likelihoods.

![Image of MultiRC example]
There have been many organisms that have lived in Earth’s past. Only a tiny number of them became fossils. Fossils provide evidence about life on Earth. They tell us that life on Earth has changed over time. Fossils in younger rocks look like animals and plants that are living today. Fossils in older rocks are less like living organisms. Fossils can tell us about where the organism lived. Was it land or marine? Fossils can even tell us if the water was shallow or deep. Fossils can even provide clues to ancient climates.

What are three things scientists learn from fossils?

Who lived in prehistoric times.
**e-SNLI example.** Groundtruth & Prediction: “neutral” (confidence: 1.00)

| P | Woman in a dress standing in front of a line of a clothing line, with clothes hanging on the line. |
| H | Her dress is **dark blue**. |
| PM | Woman in a dress standing in front of a line of a clothing line, with clothes hanging on the line. |
| HM | Her dress is dark blue. |
| IoU: 0.00, precision: 0.00, recall: 0.00 |
| PL00 | Woman in a dress standing in front of a line of a clothing line, with clothes hanging on the line. |
| HL00 | Her dress is **dark blue**. |
| IoU: 0.33, precision: 0.33, recall: 1.00 |

Figure A9: The set of **explanatory words** given by LOO_{empty} covers all highlights (higher precision and IoU) that are important to **human** in the e-SNLI “neutral” example which contains a pair of premise (P) and hypothesis (H) while there are **none** tokens given by IM are in correlation with human explanations.

---

**MultiRC example.** Groundtruth & Prediction: “True” (confidence: 0.90)

| P | There have been many organisms that have lived in Earth’s past. Only a tiny number of them became fossils. Still, scientists learn a lot from fossils. Fossils are our best clues about the history of life on Earth. Fossils provide evidence about life on Earth. They tell us that life on Earth has changed over time. Fossils in younger rocks look like animals and plants that are living today. Fossils in older rocks are less like living organisms. Fossils can tell us about where the organism lived. Was it land or marine? Fossils can even tell us if the water was shallow or deep. Fossils can even provide clues to ancient climates. |
| Q | What happened to some organisms that lived in Earth’s past? |
| A | They became fossils. Others did not become fossils |
| PM | There have been many organisms that have lived in Earth’s past. Only a tiny number of them became fossils. Still, scientists learn a lot from fossils. Fossils are our best clues about the history of life on Earth. Fossils provide evidence about life on Earth. They tell us that life on Earth has changed over time. Fossils in younger rocks look like animals and plants that are living today. Fossils in older rocks are less like living organisms. Fossils can tell us about where the organism lived. Was it land or marine? Fossils can even tell us if the water was shallow or deep. Fossils can even provide clues to ancient climates. |
| QM | What happened to some organisms that lived in Earth’s past? |
| AM | They became fossils. Others did not become fossils |
| IoU: 0.16, precision: 0.50, recall: 0.19 |
| PL00 | There have been many organisms that have lived in Earth’s past. Only a tiny number of them became fossils. Still, scientists learn a lot from fossils. Fossils are our best clues about the history of life on Earth. Fossils provide evidence about life on Earth. They tell us that life on Earth has changed over time. Fossils in younger rocks look like animals and plants that are living today. Fossils in older rocks are less like living organisms. Fossils can tell us about where the organism lived. Was it land or marine? Fossils can even tell us if the water was shallow or deep. Fossils can even provide clues to ancient climates. |
| QL00 | What happened to some organisms that lived in Earth’s past? |
| AL00 | They became fossils. Others did not become fossils |
| IoU: 0.86, precision: 0.57, recall: 0.95 |

Figure A10: The set of **explanatory words** given by LOO_{empty} covers 95% of human highlights with higher precision and IoU in the MultiRC “True” example which contains a triplet of paragraph (P), question (Q) and answer (A) while there are only few tokens given by IM are in correlation with human explanations.
There have been many organisms that have lived in Earth's past. Only a tiny number of them became fossils. Still, scientists learn a lot from fossils. Fossils are our best clues about the history of life on Earth. Fossils provide evidence about life on Earth. They tell us that life on Earth has changed over time. Fossils in younger rocks look like animals and plants that are living today. Fossils in older rocks are less like living organisms. Fossils can tell us about where the organism lived. Was it land or marine? Fossils can even tell us if the water was shallow or deep. Fossils can even provide clues to ancient climates.

What is a major difference between younger fossils and older fossils?

Older rocks are rougher and thicker than younger fossils.

---

Enormously entertaining for moviegoers of any age.

---

Figure A11: The set of explanatory words given by LOO_empty covers two thirds of human highlights with higher precision and IoU in the MultiRC "False" example which contains a triplet of paragraph (P), question (Q) and answer (A) while there are two tokens given by IM are in correlation with human explanations.

Figure A12: When a word is removed, the predicted labels of all resulting sentences (S1 to S7) are still "positive" with a confidence score of 1.0.
<table>
<thead>
<tr>
<th>e-SNLI example. Groundtruth: “entailment”</th>
<th>Prediction</th>
</tr>
</thead>
<tbody>
<tr>
<td>P Two women having <strong>drinks</strong> and smoking <strong>cigarettes</strong> at the bar.</td>
<td>entailment</td>
</tr>
<tr>
<td>H Two women are at a <strong>bar</strong>.</td>
<td>(0.99)</td>
</tr>
<tr>
<td>P₁ Two women having drinks and smoking cigarettes at the bar.</td>
<td>entailment</td>
</tr>
<tr>
<td>H₁ Two women are at a bar.</td>
<td>(0.98)</td>
</tr>
<tr>
<td>P₂ Two <strong>women</strong> having drinks and smoking cigarettes at the bar.</td>
<td>neutral</td>
</tr>
<tr>
<td>H₂ Two women are at a bar.</td>
<td>(0.93)</td>
</tr>
<tr>
<td>P₃ Two women <strong>having</strong> drinks and smoking cigarettes at the bar.</td>
<td>entailment</td>
</tr>
<tr>
<td>H₃ Two women are at a bar.</td>
<td>(0.99)</td>
</tr>
<tr>
<td>P₄ Two women having <strong>drinks</strong> and smoking cigarettes at the bar.</td>
<td>entailment</td>
</tr>
<tr>
<td>H₄ Two women are at a bar.</td>
<td>(0.99)</td>
</tr>
<tr>
<td>P₅ Two women having drinks and smoking cigarettes at the bar.</td>
<td>entailment</td>
</tr>
<tr>
<td>H₅ Two women are at a bar.</td>
<td>(0.99)</td>
</tr>
<tr>
<td>P₆ Two women having drinks and <strong>smoking</strong> cigarettes at the bar.</td>
<td>entailment</td>
</tr>
<tr>
<td>H₆ Two women are at a bar.</td>
<td>(0.99)</td>
</tr>
<tr>
<td>P₇ Two women having drinks and smoking <strong>cigarettes</strong> at the bar.</td>
<td>entailment</td>
</tr>
<tr>
<td>H₇ Two women are at a bar.</td>
<td>(0.99)</td>
</tr>
<tr>
<td>P₈ Two women having drinks and smoking cigarettes at the bar.</td>
<td>entailment</td>
</tr>
<tr>
<td>H₈ Two women are at a bar.</td>
<td>(0.98)</td>
</tr>
<tr>
<td>P₉ Two women having drinks and smoking cigarettes at the <strong>bar</strong>.</td>
<td>entailment</td>
</tr>
<tr>
<td>H₉ Two women are at a bar.</td>
<td>(0.98)</td>
</tr>
<tr>
<td>P₁₀ Two women having drinks and smoking cigarettes at the bar.</td>
<td>entailment</td>
</tr>
<tr>
<td>H₁₀ Two women are at a bar.</td>
<td>(0.97)</td>
</tr>
<tr>
<td>P₁₁ Two women having drinks and smoking cigarettes at the bar.</td>
<td>entailment</td>
</tr>
<tr>
<td>H₁₁ Two women are at a bar.</td>
<td>(0.99)</td>
</tr>
<tr>
<td>P₁₂ Two women having drinks and smoking cigarettes at the bar.</td>
<td>entailment</td>
</tr>
<tr>
<td>H₁₂ Two women are at a bar.</td>
<td>(0.99)</td>
</tr>
<tr>
<td>P₁₃ Two women having drinks and smoking cigarettes at the bar.</td>
<td>entailment</td>
</tr>
<tr>
<td>H₁₃ Two <strong>women</strong> are at a bar.</td>
<td>(0.98)</td>
</tr>
<tr>
<td>P₁₄ Two women having drinks and smoking cigarettes at the bar.</td>
<td>entailment</td>
</tr>
<tr>
<td>H₁₄ Two <strong>women</strong> are at a bar.</td>
<td>(0.99)</td>
</tr>
<tr>
<td>P₁₅ Two women having drinks and smoking cigarettes at the bar.</td>
<td>entailment</td>
</tr>
<tr>
<td>H₁₅ Two women are at a <strong>bar</strong>.</td>
<td>(0.84)</td>
</tr>
<tr>
<td>P₁₆ Two women having drinks and smoking cigarettes at the bar.</td>
<td>entailment</td>
</tr>
<tr>
<td>H₁₆ Two women are at a bar.</td>
<td>(0.97)</td>
</tr>
<tr>
<td>P₁₇ Two women having drinks and smoking cigarettes at the bar.</td>
<td>entailment</td>
</tr>
<tr>
<td>H₁₇ Two women are at a <strong>bar</strong>.</td>
<td>(0.54)</td>
</tr>
<tr>
<td>P₁₈ Two women having drinks and smoking cigarettes at the bar.</td>
<td>entailment</td>
</tr>
<tr>
<td>H₁₈ Two women are at a bar.</td>
<td>(0.95)</td>
</tr>
</tbody>
</table>

Figure A13: The removal of each token in both premise and hypothesis in e-SNLI example which contains a pair of premise (P) and hypothesis (H) **infrequently change the prediction**. Specifically, only the example of (P₂, H₂) shifted its prediction to “neutral” while the remaining partially-removed examples do not change their original prediction with high confidence score in parentheses.