Interpretation of NLP Models with Input Marginalization

Ronak Mehta
CSE 517 - Natural Language Processing
University of Washington
ronakdm@uw.edu

Mayura Patwardhan
CSE 517 - Natural Language Processing
University of Washington
mp97@uw.edu

Peter Michael
CSE 517 - Natural Language Processing
University of Washington
petermic@uw.edu

Xinyu Ma
CSE 517 - Natural Language Processing
University of Washington
xinyuma@uw.edu

Abstract

Many methods have been proposed to interpret the predictions of neural language models. One interpretation involves measuring the attribution scores, or how much each token contributes to the final prediction. Replacing each token with a predefined value, or removing the token altogether leads to a misleading interpretation due to an out of distribution problem. Kim et al. 2020 propose to solve this problem by marginalizing each token. In this paper we will reproduce from scratch the results obtained from Kim et al, by training 4 NLP models for sentiment analysis and inference. We also provide improvements on the current model including Monte Carlo sampling and next sentence prediction in order to provide a more robust analysis of interpretation. Our implementation is available on GitHub.

1 Introduction

Modern machine learning and NLP algorithms have seen a meteoric rise in the ability to achieve high accuracy on tasks involving structured, high-dimensional data such as images and text. Doing so can often come at the cost of interpretability. A common approach in NLP is to assign attribution scores to particular tokens in a sequence to determine their effect on a classification label.

Previous methods measure token attribution by replacing it with 0 then measuring the change in the correct label probability of the sequence. However, this can create an out-of-distribution (OOD) problem, deviating form the target models data distribution, as such zero-padded sequences do not exist in natural language.

Kim et al. [2020] propose to marginalize each token out to mitigate the OOD problem of the existing erasure scheme. This measures the contribution of the token over the expected value over all possible tokens. The probability of candidate tokens are given by the BERT module trained on masked-language-modeling. This method can also efficiently compute multi-token attribution scores by the same principle. A dataset-level score, called AUC_{rep} is also given by the authors.

We hypothesize that the new interpretation method of input marginalization using MLM more accurately captures the importance of tokens than the existing erasure scheme than that of zero erasure or using a constant baseline such as '[UNK]', by assigning low attribution scores to unimportant tokens and correctly highlighting the important tokens. The AUC_{rep} metric is used to quantitively measure this difference.

Preprint. Under review.

2 Scope of Reproducibility

We have three major hypotheses that are tested in this work.

1. The attribution scores of input marginalization will agree with human intuition about which tokens are important.
2. Input marginalization using MLM more accurately captures the importance of tokens than the existing erasure scheme than that of zero erasure or using a constant baseline such as '[UNK]'.
3. Input marginalization principle can be generalized for tasks other than classification.

3 Methodology

We first give a precise description of both the methods and performance metrics from the original paper, as well as our proposed extensions. We then describe exact hyperparameters and other experimental settings.

3.1 Formal description of proposed methods

We mitigate the OOD issue using input marginalization, and evaluate it quantitatively to zero-erasure and "UNK" erasure using the AUC$_{\text{rep}}$ metric proposed in the paper. Finally, we extend this methods for a new task: generative language modeling.

3.1.1 Weight-Of-Evidence Attribution Score for Classification

Weight of evidence is an attribution score for tokens used to measure changes in the model output by virtue of those tokens.

Let $x = (x_1, ..., x_n)$ be a sentence with vocabulary $\mathcal{V}$ and y be a class label. Let $p_\theta(y|x)$ be a model that predicts probability of class labels given an input, parametrized by θ. Define

$$p_\theta^{[i]}(y|x) = \sum_{w\in\mathcal{V}} p(y|x_{1:i-1}, w, x_{i+1:T}) \cdot p(w \mid x_{1:i-1}, x_{i+1:T})$$

This can also be written as

$$\log p_\theta^{[i]}(y|x) = \text{LogSumExp}\left([\log p(y|x_{1:i-1}, w, x_{i+1:T}) + \log p(w \mid x_{1:i-1}, x_{i+1:T})]_{w\in\mathcal{V}}\right),$$

The term inside the “LogSumExp" function is the vector generated by computing the term for each $w \in \mathcal{V}$. We hope to do this in parallel, and without major numerical issues. The probability $p(w \mid x_{1:i-1}, x_{i+1:T})$ can be given by masking token x_i, and calling the masked-language-model (MLM) of the pretrained BERT model. The log-probability $\log p(y|x_{1:i-1}, w, x_{i+1:T})$ can theoretically be computed in parallel by generating a $|\mathcal{V}|$-sized batch by duplicating x replacing each instance of x_i with different w's, although sending batches of words is more realistic for large vocabulary sizes. The log-odds ratio is given by

$$\begin{aligned}\text{logodds}\,(p_\theta(y|x)) &= \log \frac{p_\theta(y|x)}{1 - p_\theta(y|x)}\\ &= \log p_\theta(y|x) - \log(1 - \exp \log p_\theta(y|x))\end{aligned}$$

This is exactly given by the logits of the $p_\theta(y|x)$ model evaluated at the target class minus the logits evaluated at the non-target class. Here, target class means the class for which we want to measure the token's contribution. Finally, the weight-of-evidence attribution score a is given by

$$a^{[i]}(x) = \text{logodds}\,(p_\theta(x)) - \text{logodds}\left(p_\theta^{[i]}(x)\right)$$

3.1.2 Input Marginalization

The log odds probability in the weight of evidence is calculated using input marginalization.

$p(y_c \mid x_{1:i-1}, x_{i+1:T})$, the MLM probability, can be rewritten as:

$$p(w \mid x_{1:i-1}, x_{i+1:T}) = \sum_{\widetilde{x_i} \in V} p(y_c, \widetilde{x_i} | x_{1:i-1}, x_{i+1:T})$$

$$= \sum_{\widetilde{x_i} \in V} p(y_c | \widetilde{x_i}, x_{1:i-1}, x_{i+1:T}) p(\widetilde{x_i} | x_{1:i-1}, x_{i+1:T})$$

where $\widetilde{x_i}$ is the candidate token we are replacing at position i. $p(y_c|\widetilde{x_i}, x_{1:i-1}, x_{i+1:T})$ is interpreted as the probability of the target class y_c given the input sentence replaced by $\widetilde{x_i}$ at position i. It is calculated using the network to be interpreted. We compute $p(\widetilde{x_i}|x_{1:i-1}, x_{i+1:T})$ using BERT MLM.

3.1.3 Truncated Input Marginalization

Marginalizing over the entire vocabulary (over 30,000 tokens) is computationally expensive. Instead, we index tokens that have an MLM likelihood greater than a specified threshold, and compute the marginalization over only those tokens (renormalizing the probabilities).

3.1.4 Deletion Curves and AUC for Classification

In order to understand of our input marginalization method worked better than previous methods (such as zero erasure and "UNK" erasure), we used the AUC$_{\text{rep}}$ metric. The prediction probability curve is plotted with important tokens gradually getting replaced. The AUC$_{\text{rep}}$ score is the area under the curve.

Let $x_{(1)}, ..., x_{(n)}$ be the tokens of sentence, ordered such that

$$a^{[(n)]}(x) \geq a^{[(n-1)]}(x) \geq \cdots \geq a^{[(1)]}(x).$$

That is, $x_{(n)}$ has the largest attribution score and $x_{(1)}$ the smallest. Let

$$x^{[i]} = x \text{ with } x_{(1)}, x_{(2)}, ..., x_{(i)} \text{ replaced with <PAD>}$$

Let $\hat{y} = \arg\ \max_y p_\theta(y|x)$. The deletion curve for model $p_\theta(\hat{y}|x^{[i]})$ against $i = 1, ..., n$. The curve is meant to drop rapidly to indicate that the attribution scores are faithful.

3.2 Model Description

Three NLP models to be trained and fine-tuned for sentiment analysis and natural language inference for the proposed method. For all models, the embedding dimension for embedding layer was set to 100, with BERT tokenizer vocabulary set, and activation function is ReLU. The output dimension is two for SST-2 and three for SNLI.

- **CNN for SST-2**. An 1-dimensional convolutional neural networks consists of an embedding layer, three convolution layers and a fully connected (FC) layer. 100 filters with size three, four and five are used for the convolution layers.The dropout rate for the convolution layer is set to 0.5. We use a cross-entropy loss. it has 3,360,601 parameters.
- **LSTM for SST-2**. A bidirectional long short-term memory (LSTM) comprises an embedding layer, two bidirectional LSTM layers with a hidden dimension of 200 and a FC layers. The dropout rate for the embedding layer, LSTM, and the fully connected layer was set to 0.3, 0.5, and 0.5, respectively. We use a cross-entropy loss. It has 4503403 parameters.
- **LSTM for SNLI**. This model involves an embedding layer, a projection layer, a bidirectional LSTM layer, and four FC layers. The projection layer is an FC layer with an output dimension of 300. The encoder consists of one bidirectional LSTM layer with a hidden dimension of 300. Both premise and hypothesis are encoded with the same encoder and concatenated before the FC layer. We use a cross-entropy loss. It has 4503403 parameters.
- **BERT for SST-2**. `BERTForSentenceClassification` with pre-trained weights was fine-tuned for two epochs. We use a cross-entropy loss.
- **Transformer for Wikitext-2** Neural autoregressive model with 8 transformer block layers, each with 8 attention heads. The word embedding dimension is 200, the key/query/value dimension is 40, and the hidden state dimension is 500. We use a cross-entropy loss. The model has 9824721 parameters.

3.3 Data Description

We use 3 datasets. The first dataset is the Stanford Sentiment Treebank 2 (SST-2) is a dataset for predicting sentiment from 11,855 longer movie reviews, annotated by critics. The second dataset, Stanford Natural Language Inference (SNLI), is a dataset that contains human-written sentences that are manually annotated with how sentence pairs relate to each other (e.g., contradiction). In both cases, labels are evenly split among classes. It has 570k pairs. The last dataset is Wikitext-2, an unstructured language modeling dataset extracted from articles on Wikipedia with over 2 million training tokens. We believe that these datasets provide a diverse set of tasks on which to thoroughly evaluate the input marginalization technique. We used official train/validation/test splits for these datasets.

3.4 Hyperparameters

See 3.2 for number of hidden states of each model.

- **CNN for SST-2**. Learning rate: 0.0001, dropout: 0.5, batch size: 10, epochs: 30.
- **LSTM for SST-2**. Learning rate: 0.0001, dropout: 0.5, batch size: 10, epochs: 30.
- **LSTM for SNLI**. Learning rate: 0.0001, dropout for training: 0.5, dropout for embedding: 0.3, batch size: 128, epochs: 30.
- **BERT for SST-2**. Learning rate: 0.00002, dropout: 0.5, batch size: 32, epochs: 2.
- **Transformer for Wikitext-2** Learning rate: 0.0001, dropout: none, batch size: 32, epochs: 15, context length 150.

3.5 Code

We implemented the result from the paper from scratch, including fitting the models, implementing the metrics, and visualizaing the colored sentences. For the generative models task, we adapted code from a homework assignment from the Generative Models course at UW to train the transformer, but implemented a novel attribution score method ourselves. The code and instructions can be found on GitHub.

4 Computational Requirements

We used the standard Google Colab GPU for training all of our models. We did 1 trial per model.

Training time:

- **CNN for SST-2**. Total time: $4:08$. Average epoch time: $\sim 0:08$.
- **LSTM for SST-2**. Total time: $15:02$. Average epoch time: $\sim 0:29$.
- **LSTM for SNLI**.
- **BERT for SST-2**. Total time: $3:15$. Average epoch time: $\sim 1:37$.
- **Transformer for Wikitext-2** Total time: $39:30$. Average epoch time: $\sim 3:26$.

5 Results

5.1 Reproducibility results

5.1.1 Model Training

In order to evaluate the method in both a model agnostic and task agnostic way, 3 different models were trained in two different tasks. These models are used in the to calculate the probabilities in the input marginalization. The test accuracy of these target models is provided in Table 1. We were able to obtain test accuracy on par with the original paper.

Table 1: Test accuracy of the target models.

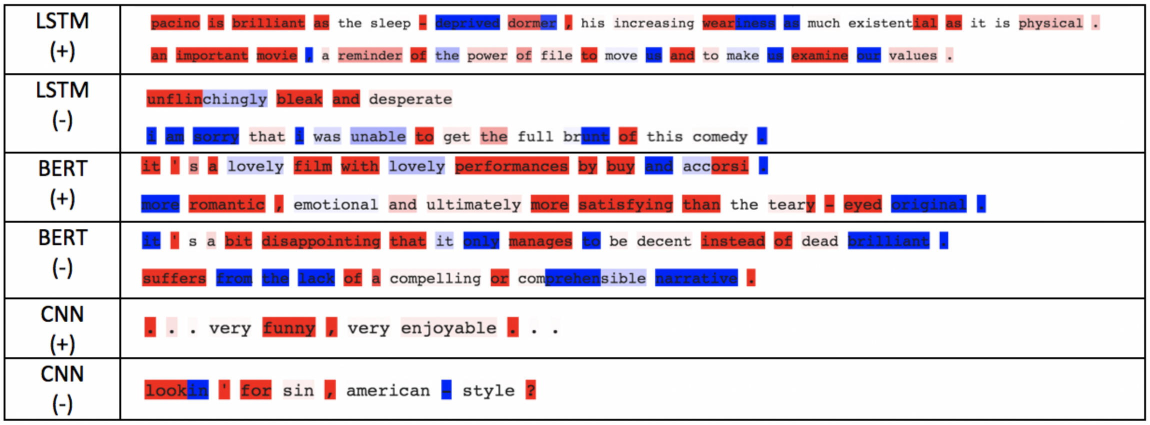
Corpus	LSTM	BERT	CNN
SST-2	0.77	0.92	0.75
SNLI	0.67	–	–

5.1.2 Interpretation Results

The results of the attribution scores are represented by the colored sentences in Figure 1. The color of the token indicates the contribution of that token to the final prediction. Red represents a positive contribution, while blue represents a negative contribution. The magnitude of the contribution is reflected in the intensity is color.

The results agree with Hypothesis 1 that the scores of the input marginalization agree with the human intuition. Each model was trained on both positively labeled and negatively labeled sentences. In our model, we can see that words such as "brilliant" and "romantic" are colored red in the positive sentences. While words such as "deprived" are colored blue. Similarly, we can see that for the negative class "bleak" and "disappointing" are colored red, while "brilliant" is colored blue. This is on par with what we would expect intuitively. However, there were some deviations from this pattern, which could have been caused by sparsity of certain words in the training data.

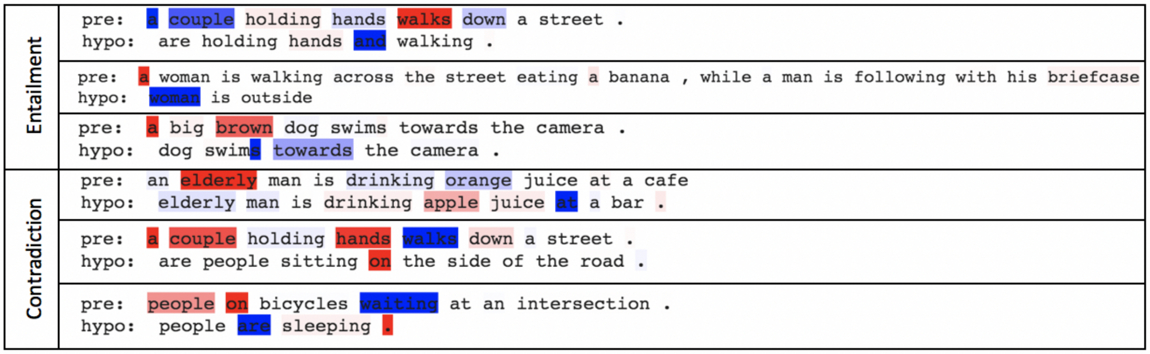
Our results generally followed the results obtained from the original paper (Figure 5) with some deviations. This could have been caused by minor architectural differences in our models. In addition, the color scale has some slight variation from the original paper, causing some differences in the visualization.



LSTM (+)	pacino is brilliant as the sleep - deprived dormer , his increasing weariness as much existential as it is physical . an important movie , a reminder of the power of file to move us and to make us examine our values .
LSTM (-)	unflinchingly bleak and desperate i am sorry that i was unable to get the full brunt of this comedy .
BERT (+)	it ' s a lovely film with lovely performances by buy and accorsi . more romantic , emotional and ultimately more satisfying than the teary - eyed original .
BERT (-)	it ' s a bit disappointing that it only manages to be decent instead of dead brilliant . suffers from the lack of a compelling or comprehensible narrative .
CNN (+)	. . . very funny , very enjoyable . . .
CNN (-)	lookin ' for sin , american - style ?

Figure 1: Interpretation Results for SST-2

The interpretation for the LSTM for the SNLI are similar to that of Figure 1. The red and blue colors represent the tokens attribution to the given class. Our results show some significant differences between the attribution scores that we obtained compared to what the original paper obtained (Figure 2). Like the paper, the sentences were correctly classified to the denoted class.



Entailment

pre: a couple holding hands walks down a street .
hypo: are holding hands and walking .

pre: a woman is walking across the street eating a banana , while a man is following with his briefcase
hypo: woman is outside

pre: a big brown dog swims towards the camera .
hypo: dog swims towards the camera .

Contradiction

pre: an elderly man is drinking orange juice at a cafe
hypo: elderly man is drinking apple juice at a bar .

pre: a couple holding hands walks down a street .
hypo: are people sitting on the side of the road .

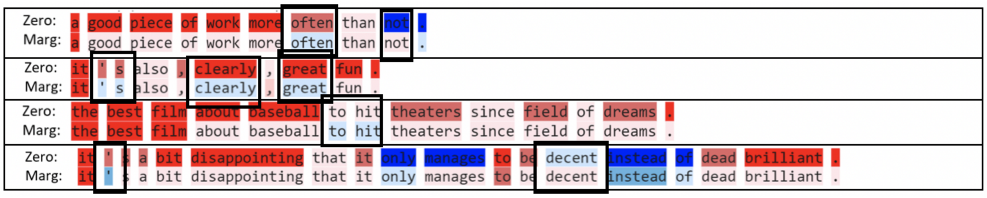
pre: people on bicycles waiting at an intersection .
hypo: people are sleeping .

Figure 2: Interpretation Results for SNLI

5.1.3 Comparison To Existing Scheme

The results agree with Hypothesis 2 that input marginalization better captures the importance of tokens than the existing erasure scheme. The comparison results using zero erasure (Zero) and input marginalization (Marg) are shown in Figure 3. With all sentences were classified as positive, zero erasure is often assigning high attrition score to uninformative tokens such as punctuation and "of". The proposed input-marginalization method clearly showed unimportant tokens were given low attribution scores and is able to correctly highlight the important ones.

Our comparison also had some discrepancies that are important to note. For example, the word "great" in a positive sentence should be labeled as red since contributes to the classification of the token. However, it is labeled as blue in the input marginalization, providing evidence that the input marginalization does not work flawlessly in our test cases.



Zero: a good piece of work more often than not .
Marg: a good piece of work more often than not .

Zero: it ' s also , clearly , great fun .
Marg: it ' s also , clearly , great fun .

Zero: the best film about baseball to hit theaters since field of dreams .
Marg: the best film about baseball to hit theaters since field of dreams .

Zero: it ' s a bit disappointing that it only manages to be decent instead of dead brilliant .
Marg: it ' s a bit disappointing that it only manages to be decent instead of dead brilliant .

Figure 3: Interpretations Results comparing zero erasure and input marginalization

We also used the proposed method AUC$_{\text{rep}}$ to compare the input marginalization with zero-erasure and another baseline which use "[UNK]" and verified that the OOD problem still existed no matter what predefined character used.The deletion curve on Figure 4 showed the change in prediction probabilities as token with high attribution score are gradually replaced. The deletion curve showed that with using input-marginalization, the prediction probability drops the most rapidly, which implies that the proposed method interpret the important tokens better compared to the zero and "[UNK]" erasures.

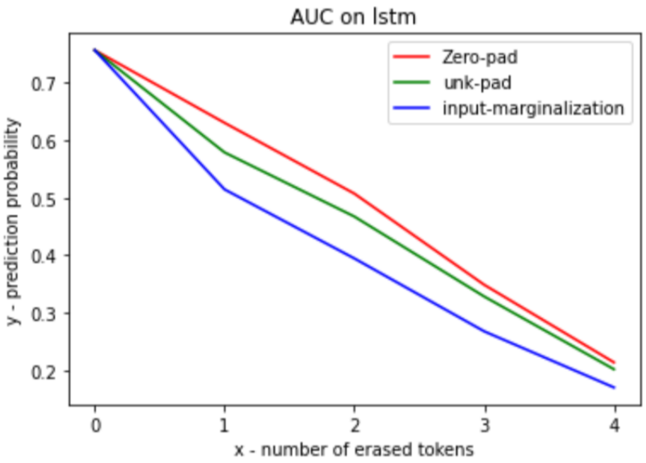


Figure 4: AUC Deletion Curve for LSTM

The average AUC_{rep} values for 700 SST-2 sentences are provided in Table 2, with our method showing the lowest AUC_{rep}, demonstrating input marginalization captures the importance of tokens more accurately than the existing erasure scheme. But since we were not able to use the same criterion to color the sentence, and the performance of the model we used are a little different than their model (Figure 7), small deviation existed between our results and the results from the paper.

Table 2: Comparison of AUC_{rep} with the existing erasure scheme.

Interpretation Method	Zero	Unk	Ours
AUC_{rep}	0.5608	0.5328	0.4834

5.2 Experiments beyond the original paper

A major follow-on experiment of our project is to extend the principle of input marginalization to a completely different task. We derive a similar method for the generative modeling task and show proof-of-concept results. A general theme for extending the method to language modeling has been to replace $p_\theta(y|x)$ with $p_\theta(x)$ where possible.

5.2.1 Formal description

As before, let $x = (x_1, ..., x_n$ represent a sentence in vocabulary $\mathcal{V}$. Assume access to $\log p_\theta(x)$, which is the output of the generative model to be evaluated.

$$\begin{aligned}
p_\theta^{[i]}(x) &= \sum_{w \in \mathcal{V}} p(x_{1:i-1}, w, x_{i+1:T}) \cdot p(w \mid x_{1:i-1}, x_{i+1:T}) \\
\log p_\theta^{[i]}(x) &= \log \sum_{w \in \mathcal{V}} p(x_{1:i-1}, w, x_{i+1:T}) \cdot p(w \mid x_{1:i-1}, x_{i+1:T}) \\
&= \log \sum_{w \in \mathcal{V}} \exp \left\{ \log p(x_{1:i-1}, w, x_{i+1:T}) + \log p(w \mid x_{1:i-1}, x_{i+1:T}) \right\} \\
&= \text{LogSumExp} \left([\log p(x_{1:i-1}, w, x_{i+1:T}) + \log p(w \mid x_{1:i-1}, x_{i+1:T})]_{w \in \mathcal{V}} \right)
\end{aligned}$$

$p(w|x_{1:i-1}, x_{i+1:T})$ is given by the masked language model (MLM) of BERT, and $p(x_{1:i-1}, w, x_{i+1:T})$ can be computed by the generative model (by replacing the i-th token of x with word w). Once this is computed, we can compute

$$\text{logodds}\left(p_\theta^{[i]}(x)\right) = \log p_\theta^{[i]}(x) - \log(1 - \exp \log p_\theta^{[i]}(x))$$

Finally, the attribution scores $a(x)$ are given by

$$a^{[i]}(x) = \text{logodds}\left(p_\theta(x)\right) - \text{logodds}\left(p_\theta^{[i]}(x)\right)$$

5.2.2 Interpretation results

See Sections 3.2 and 3.4 for precise descriptions and hyperparameters of the generative model being evaluated (in this case, a transformer network). Because of the computational overhead of the generative models method, we were only able to evaluate it on very short sentences (see Section 6). The colored sentences are below. One of the most defining words of the sentence, the proper noun "sam", has a high attribution score indicating a large contribution to the likelihood assigned by the model. This confirms Hypothesis 3 at least in theory, with computational considerations being the next major question.

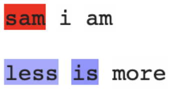


Figure 5: Sentences colored by attribution score to likelihood assigned by transformer model.

6 Discussion

To review, our hypothesis was that input marginalization would be able to more accurately capture the importance of tokens than existing erasure schemes. From the results of our interpretation word colorings, we are able to visualize that the attribution scores given by our input marginalization model are able to accurately predict words that contribute to a classification model class. In addition, when comparing to zero erasure, we can see that zero erasure often assigned high attribution scores to uninformative tokens such as "to" and ",", which the input marginalization was able to avoid. However, we ran into some discrepancies in our input marginalization that need to be further investigated in future research. These results are quantitatively validated in part two of our experiment, where we used that AUC$_{\text{rep}}$ score to show that the deletion curves in the input marginalization dropped more rapidly when compared to zero and UNK erasures. This provides evidence that our method better captures the relative importance of tokens.

Overall we were able to replicate many important results that were captured in the paper. However, because of variations in our model, as well as variations in our color scheme, we obtained some results that differed.

We also presented a methodological extension of the method to the generative models task. In principle, the nature of our generalization could be applied to any task that scores sentences or sentence-label pairs. One difficulty in the generative models case is that our neural autoregressive model had to be called many times in order to compute the attribution scores for one sentence. If the sentence has T tokens, the model requires T calls to get the probability of a sequence. Considering marginalizing over the entire vocabulary $\mathcal{V}$, this takes $O(|\mathcal{V}|T^2)$ calls, which is computationally taxing. Future work can explore more efficient or approximate methods.

What was easy: The input marginalization was generally pretty simple and easy to understand. Surprisingly training the models was much easier than implementing the deletion curves and input marginalization. This is likely due to the highly standard nature of training classification models.

What was difficult: We had some trouble with implementing the input marginalization, and had to adjust our code to work for BERT, as well as our 3 other models. Finally, we ran into some subtle bugs when computing our AUC curve that took time to identify and solve.

Recommendations: One recommendation we have to the authors is having a more systematic way of interpreting some of their figures. The colors are somewhat arbitrary and can lead to misinterpretations if not careful. Another recommendation we have is to have better supporting documentation for some of the methods they used. We specifically felt the section about the AUC metric could have been better detailed.

7 Appendix - Original Results from the Paper

7.1 Model Training

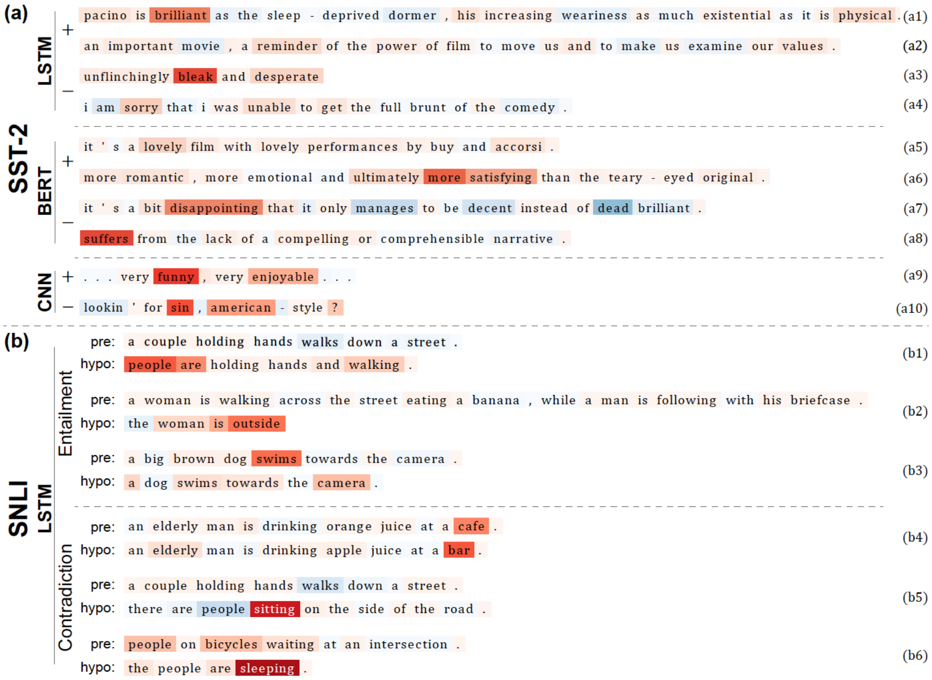
```
Algorithm 1 Input marginalization
  Input Target model $\theta$, input $\boldsymbol{x}$, vocabulary $\mathcal{V}$,
  likelihood threshold $\sigma$, and target class $y_c$
  Output Attribution score $\boldsymbol{a}$
  for $i = 0$ to length($\boldsymbol{x}$) do
      $m \leftarrow 0$                ▷ Initialize attribution score
      $\boldsymbol{s} \leftarrow$ copy $\boldsymbol{x}$
      $\boldsymbol{s}_i \leftarrow$ "[MASK]" token
      for all $\tilde{s}_i$ in $\mathcal{V}$ do
          $p(\tilde{s}_i|\boldsymbol{s}_{-i}) \leftarrow \text{BERT}_{\text{MLM}}(\boldsymbol{s})$
          if $p(\tilde{s}_i|\boldsymbol{s}_{-i}) > \sigma$ then
              $\boldsymbol{s}_i \leftarrow \tilde{s}_i$
              $m \leftarrow m + p(\tilde{s}_i|\boldsymbol{s}_{-i}) \cdot p_\theta(y_c|\boldsymbol{s})$
          end if
      end for
      $\boldsymbol{a}_i = \text{logodds}_\theta(y_c|\boldsymbol{x}) - |\text{logodds}_\theta(m)$
                  ▷ Prediction difference measurement
  end for
```

Figure 6: Input marginalization algorithm

Table 3: Test accuracy of the target models

Corpus	LSTM	BERT	CNN
SST-2	0.7753	0.8578	0.7300
SNLI	0.6314	–	–

7.2 Interpretation Results



pacino is brilliant as the sleep - deprived dormer , his increasing weariness as much existential as it is physical . (a1)

an important movie , a reminder of the power of film to move us and to make us examine our values . (a2)

unflinchingly bleak and desperate (a3)

i am sorry that i was unable to get the full brunt of the comedy . (a4)

it ' s a lovely film with lovely performances by buy and accorsi . (a5)

more romantic , more emotional and ultimately more satisfying than the teary - eyed original . (a6)

it ' s a bit disappointing that it only manages to be decent instead of dead brilliant . (a7)

suffers from the lack of a compelling or comprehensible narrative . (a8)

. . . very funny , very enjoyable . . . (a9)

lookin ' for sin , american - style ? (a10)

pre: a couple holding hands walks down a street .
hypo: people are holding hands and walking . (b1)

pre: a woman is walking across the street eating a banana , while a man is following with his briefcase .
hypo: the woman is outside (b2)

pre: a big brown dog swims towards the camera .
hypo: a dog swims towards the camera . (b3)

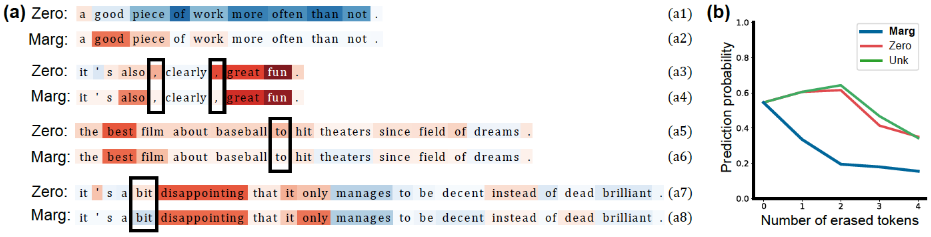
pre: an elderly man is drinking orange juice at a cafe .
hypo: an elderly man is drinking apple juice at a bar . (b4)

pre: a couple holding hands walks down a street .
hypo: there are people sitting on the side of the road . (b5)

pre: people on bicycles waiting at an intersection .
hypo: the people are sleeping . (b6)

Figure 2: Interpretation results of the proposed method. "+" and "-" in (a) denote the positive and negative classes of the depicted sentences. "pre" and "hypo" in (b) denote premise and hypothesis of SNLI, respectively. Red and blue colors denote positive and negative contributions to the denoted classes, respectively.

Figure 7: Interpretation Results



Zero: a good piece of work more often than not . (a1)
Marg: a good piece of work more often than not . (a2)

Zero: it ' s also , clearly , great fun . (a3)
Marg: it ' s also , clearly , great fun . (a4)

Zero: the best film about baseball to hit theaters since field of dreams . (a5)
Marg: the best film about baseball to hit theaters since field of dreams . (a6)

Zero: it ' s a bit disappointing that it only manages to be decent instead of dead brilliant . (a7)
Marg: it ' s a bit disappointing that it only manages to be decent instead of dead brilliant . (a8)

Figure 3: (a) shows examples of interpretations obtained by zero erasure and input marginalization (ours). Red and blue colors denote positive and negative contributions to the predicted classes, respectively. (a1-a6) are correctly classified as positive, and (a7-a8) are correctly classified to negative. (b) shows deletion curves of input marginalization, zero erasure, and "[UNK]" erasure, which are abbreviated as "Marg", "Zero", and "Unk", respectively.

Figure 8: Interpretation Results

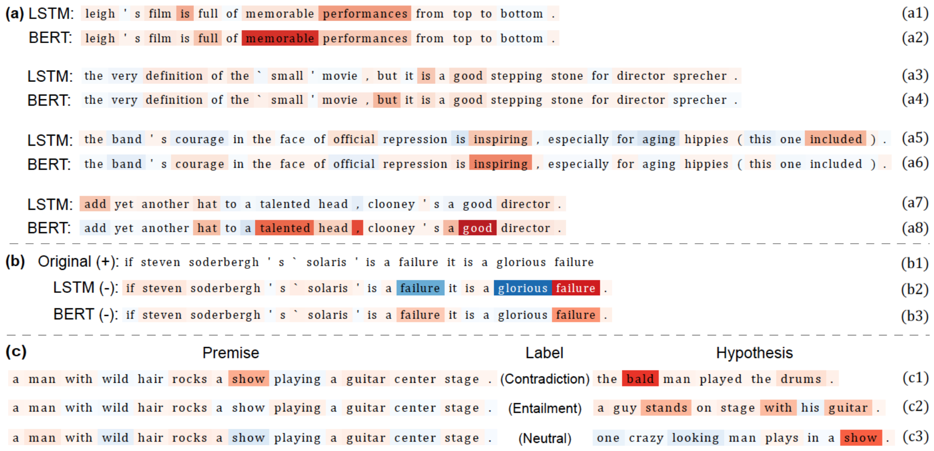

(a) LSTM: leigh ' s film is full of memorable performances from top to bottom . (a1)
BERT: leigh ' s film is full of memorable performances from top to bottom . (a2)
LSTM: the very definition of the ` small ' movie , but it is a good stepping stone for director sprecher . (a3)
BERT: the very definition of the ` small ' movie , but it is a good stepping stone for director sprecher . (a4)
LSTM: the band ' s courage in the face of official repression is inspiring , especially for aging hippies (this one included) . (a5)
BERT: the band ' s courage in the face of official repression is inspiring , especially for aging hippies (this one included) . (a6)
LSTM: add yet another hat to a talented head , clooney ' s a good director . (a7)
BERT: add yet another hat to a talented head , clooney ' s a good director . (a8)
(b) Original (+): if steven soderbergh ' s ` solaris ' is a failure it is a glorious failure (b1)
LSTM (-): if steven soderbergh ' s ` solaris ' is a failure it is a glorious failure . (b2)
BERT (-): if steven soderbergh ' s ` solaris ' is a failure it is a glorious failure . (b3)
(c) Premise Label Hypothesis
a man with wild hair rocks a show playing a guitar center stage . (Contradiction) the bald man played the drums . (c1)
a man with wild hair rocks a show playing a guitar center stage . (Entailment) a guy stands on stage with his guitar . (c2)
a man with wild hair rocks a show playing a guitar center stage . (Neutral) one crazy looking man plays in a show . (c3)

Figure 4: Interpretation results using input marginalization. Red and blue colors denote positive and negative contributions to the predicted classes. (a) shows interpretations of SST-2 predictions. (a1-a6) are correctly classified to positive, and (a7-a8) are correctly classified to negative. (b) shows positive sentences which are misclassified to negative by both LSTM and BERT. (c) shows the interpretations of SNLI predictions.

Figure 9: Interpretation Results

7.3 Comparison To Existing Scheme

Table 4: Comparison of AUC_{rep} with the existing erasure scheme (the lower the better).

Interpretation Method	Zero	Unk	Ours
AUC_{rep}	0.5284	0.5170	0.4972

References

Siwon Kim, Jihun Yi, Eunji Kim, and Sungroh Yoon. Interpretation of NLP models through input marginalization. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 3154–3167, Online, November 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.emnlp-main.255. URL https://www.aclweb.org/anthology/2020.emnlp-main.255.