Explaining Neural Network Predictions on Sentence Pairs via Learning Word-Group Masks

Hanjie Chen[1] **Song Feng**[2] **Jatin Ganhotra**[2] **Hui Wan**[2]
Chulaka Gunasekara[2] **Sachindra Joshi**[2] **Yangfeng Ji**[1]
[1]Department of Computer Science, University of Virginia, Charlottesville, VA, USA
[2]IBM Research AI
{hc9mx, yangfeng}@virginia.edu
{sfeng, jatinganhotra, hwan}@us.ibm.com
{Chulaka.Gunasekara@, jsachind@in.}ibm.com

Abstract

Explaining neural network models is important for increasing their trustworthiness in real-world applications. Most existing methods generate post-hoc explanations for neural network models by identifying individual feature attributions or detecting interactions between adjacent features. However, for models with text pairs as inputs (e.g., paraphrase identification), existing methods are not sufficient to capture feature interactions between two texts and their simple extension of computing all word-pair interactions between two texts is computationally inefficient. In this work, we propose the Group Mask (GMASK) method to implicitly detect word correlations by grouping correlated words from the input text pair together and measure their contribution to the corresponding NLP tasks as a whole. The proposed method is evaluated with two different model architectures (decomposable attention model and BERT) across four datasets, including natural language inference and paraphrase identification tasks. Experiments show the effectiveness of GMASK in providing faithful explanations to these models [1].

1 Introduction

Explaining deep neural networks is critical for revealing their prediction behaviors and enhancing the trustworthiness of applying them in real-world applications. Many methods have been proposed to explain neural network models from the post-hoc manner that generates faithful explanations based on model predictions (Ribeiro et al., 2016; Lundberg and Lee, 2017; Sundararajan et al., 2017; Guidotti et al., 2018). Most existing work focuses on identifying word attributions (Rocktäschel et al., 2015; Li et al., 2016; Thorne et al., 2019) for NLP tasks. Knowing which individual features are important might not be enough for explaining model behaviors. Then, other recent work exploits feature interactions as explanations (Singh et al., 2018; Chen et al., 2020; Tsang et al., 2020). However, they could suffer computation inefficiency while computing interactions between all word pairs, and they also fall short for identifying multiple important words correlated from different input sources for predictions. Such intuitions are particularly important for explaining sentence pair modeling tasks such as natural language inference (NLI) (Bowman et al., 2015) and paraphrase identification (PI) (Yin and Schütze, 2015).

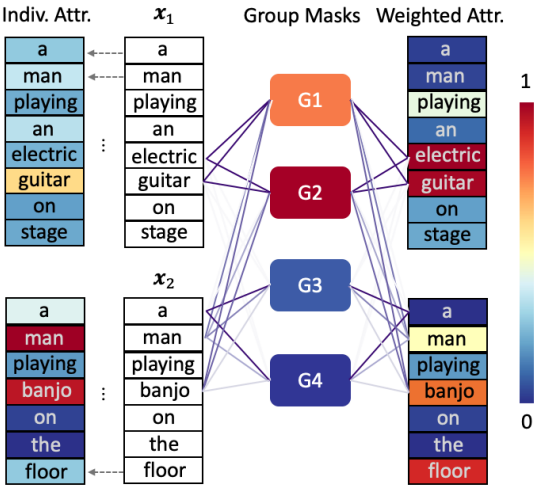


Figure 1: An illustration of obtaining individual word attributions (Indiv. Attr.) and weighted word attributions (Weighted Attr.), where the color of each block represents word importance or group importance, and the color saturation of purple lines indicates the probability of a word belonging to a specific group.

Figure 1 shows an example of NLI, where the model makes correct prediction as CONTRADICTION. The first column visualizes individual word attributions to the prediction, where the top four important words are `man`, `banjo`, `guitar`, `a`. However, the correlations between them are unclear and intuitively `man` and `a` are irrelevant to the model prediction, which makes the explanation

[1]Code for this paper is available at https://github.com/UVa-NLP/GMASK

Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pages 3917–3930
June 6–11, 2021.

untrustworthy. A good explanation should be able to capture correlated words between the sentence pair, and identify their importance to the model prediction.

In this work, we propose Group Masks (GMASK), a model-agnostic approach that considers the importance of correlated words from two input sentences. In particular, it distributes correlated words into a group and learns the group importance. In Figure 1, the input words are distributed in four groups with importance $G2 > G1 > G3 > G4$. The color saturation of purple lines represents the probability of a word belonging to a group. Different from individual word attributions, GMASK assigns `electric`, `guitar`, and `banjo` into important groups ($G2/G1$), while `man` and `a` into unimportant groups ($G3/G4$). The weighted word attributions computed as the weighted sum of group importance identify the important words `electric`, `guitar` from $\boldsymbol{x}_1$ and `banjo` from $\boldsymbol{x}_2$, which explains the model prediction.

The contribution of this work is three-fold: (1) we introduce GMASK method to explain sentence pair modeling tasks by learning weighted word attributions based on word correlations; (2) we propose a sampling-based method to solve the optimization objective of GMASK; and (3) we evaluate the proposed method with two types neural network models (decomposable attention model (Parikh et al., 2016) and BERT (Devlin et al., 2018)), for two types of sentence pair modeling tasks on four datasets. Experiments show the superiority of GMASK in generating faithful explanations compared to other competitive methods.

2 Related Work

Many approaches have been proposed to explain deep neural networks from the post-hoc manner, such as gradient-based explanation methods (Hechtlinger, 2016; Sundararajan et al., 2017), attention-based methods (Ghaeini et al., 2018; Serrano and Smith, 2019), and decomposition-based methods (Murdoch et al., 2018; Du et al., 2019). However these white-box explanation methods are either rendering doubt regarding faithfulness (Jain and Wallace, 2019; Wiegreffe and Pinter, 2019) or being limited to specific neural networks. In this work, we mainly focus on model-agnostic explanation methods, which are applicable to any black-box models.

Feature attributions Many approaches explain models by assigning feature attributions to model predictions. For example, perturbation-based methods quantify feature attributions by erasing features (Li et al., 2016) or using local linear approximation as LIME (Ribeiro et al., 2016). KernelSHAP (Lundberg and Lee, 2017) utilized Shapley values (Shapley, 1953) to compute feature attributions. Another line of work proposed learning feature attributions, such as L2X (Chen et al., 2018) which maximizes mutual information to recognize important features, and IBA (Schulz et al., 2020) which identifies feature attributions by optimizing the information bottleneck (Tishby et al., 2000). However, these approaches produce individual word attributions without considering feature correlations. Our proposed method implicitly detects correlated words and generates weighted word attributions.

Feature interactions Some work proposed to generate explanations beyond word-level features by detecting feature interactions. Murdoch et al. (2018) proposed contextual decomposition (CD) to compute word interactions and Singh et al. (2018) and Jin et al. (2019) further proposed hierarchical versions based on that. Other work adopted Shapley interaction index to compute feature interactions (Lundberg et al., 2018) and build hierarchical explanations (Chen et al., 2020). However, computing feature interactions between all word pairs is computationally inefficient (Tsang et al., 2018). Methods which only consider the interactions between adjacent features are not applicable to sentence pair modeling tasks as critical interactions usually form between words from different sentences. GMASK distributes correlated words from the input text pair into a group, and learns the group importance, without explicitly detecting feature interactions between all word pairs.

Word masks Some related work utilized word masks to select important features for building interpretable neural networks (Lei et al., 2016; Bastings et al., 2019) or improving the interpretability of existing models (Chen and Ji, 2020). De Cao et al. (2020) proposed to track the information flow of input features through the layers of BERT models. Different from the prior work, GMASK applies masks on a group of correlated words.

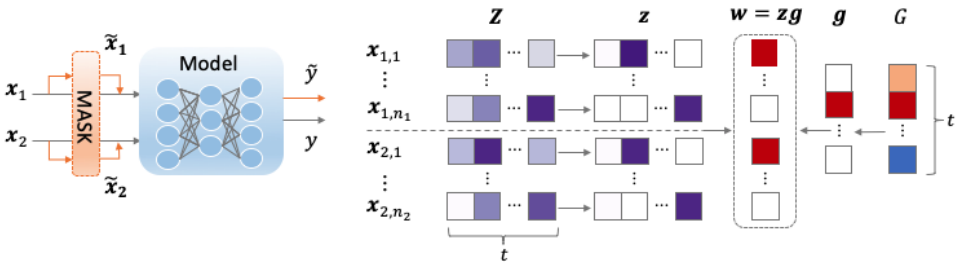


Figure 2: The left part shows that masks are applied on the word embedding layer, selecting important words for the neural network model. The outputs y and $\tilde{y}$ are corresponding to the original input $\boldsymbol{x} = [\boldsymbol{x}_1, \boldsymbol{x}_2]^T$ and masked input $\tilde{\boldsymbol{x}} = [\tilde{\boldsymbol{x}}_1, \tilde{\boldsymbol{x}}_2]^T$ respectively. The right part shows the sampling process of GMASK. For $\boldsymbol{Z}$, the color saturation of purple blocks represents the probability of a word belonging to a specific group (i.e. $\phi_{i,j}(\iota)$). $\boldsymbol{z}$ is a sample of $\boldsymbol{Z}$ with binary values. For G, the color of each block represents group importance. $\boldsymbol{g}$ is a one-hot vector sampled from G, indicating which group being selected. $\boldsymbol{w}$ is a sample of word masks obtained by multiplying $\boldsymbol{z}$ and $\boldsymbol{g}$.

3 Method

This section introduces the proposed GMASK method. GMASK implicitly learns word correlations, and distributes correlated words from different input sentences into a group. GMASK learns the importance of each group by randomly masking out groups of words. Finally, the weighted word attributions are computed based on word group distributions and group importance.

3.1 Explaining Models with Word Masks

As the left part of Figure 2 shows, the word masks are applied on input word embeddings, learning to select important words to explain the model prediction. For each input data, we generate a post-hoc explanation by learning a set of mask values which represent the word attributions.

For sentence pair modeling tasks, the input contains two sentences $\boldsymbol{x}_1 = [\boldsymbol{x}_{1,1}^T, \ldots, \boldsymbol{x}_{1,n_1}^T]^T$ and $\boldsymbol{x}_2 = [\boldsymbol{x}_{2,1}^T, \ldots, \boldsymbol{x}_{2,n_2}^T]^T$, where $\boldsymbol{x}_{i,j} \in \mathbb{R}^d$ ($i \in \{1,2\}$, $j \in \{1, \ldots, n_i\}$) represents the word embedding and n_1 and n_2 are the number of words in the two texts respectively. We denote the neural network model as $f(\cdot)$ which takes $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$ as input and outputs a prediction label $y = f(\boldsymbol{x})$, where $\boldsymbol{x} = [\boldsymbol{x}_1, \boldsymbol{x}_2]^T$. To explain the model prediction, we learn a set of word masks $\boldsymbol{W} = [W_{1,1}, \ldots, W_{1,n_1}, W_{2,1}, \ldots, W_{2,n_2}]^T$ to identify important words by multiplying the masks with input word embeddings,

$$\tilde{\boldsymbol{x}} = \boldsymbol{W} \odot \boldsymbol{x}, \tag{1}$$

where $\odot$ is an element-wise multiplication, the masked input $\tilde{\boldsymbol{x}} = [\tilde{\boldsymbol{x}}_1, \tilde{\boldsymbol{x}}_2]^T$, $\tilde{\boldsymbol{x}}_{i,j} = W_{i,j} \cdot \boldsymbol{x}_{i,j}$ ($i \in \{1,2\}$, $j \in \{1, \ldots, n_i\}$), and $W_{i,j} \in \{0,1\}$ is a binary random variable with 1 and 0 indicating to select or mask out the word $\boldsymbol{x}_{i,j}$ respectively. To generate an effective explanation, the word masks $\boldsymbol{W}$ should have the following properties: (1) correctly selecting important words for the model prediction; (2) removing as many irrelevant words as possible to keep the explanation concise; (3) selecting or masking out correlated words together from the sentence pair.

Previous work on learning individual word masks only focuses on the first two properties (Chen and Ji, 2020; De Cao et al., 2020). To satisfy the third property, We propose GMASK to implicitly detect word correlations and distribute the correlated words into a group (e.g. `electric`, `guitar`, and `banjo` are assigned to $G1$ or $G2$ in Figure 1), and learn a group mask for these words. Specifically, we decompose each $W_{i,j}$ in $\boldsymbol{W}$ into two random variables,

$$W_{i,j} = \sum_{\iota=1}^{t} \delta(Z_{i,j}, \iota)\delta(G, \iota), \tag{2}$$

where t is the predefined number of groups, and we will introduce how to pick up a t in subsection 3.3. $Z_{i,j} \in \{1, \ldots, t\}$ indicates the word $\boldsymbol{x}_{i,j}$ belonging to which group, and $G \in \{1, \ldots, t\}$ indicates which group takes the mask value 1, which means all words in this group are selected as important words, while other words in the rest groups are masked out. $\delta(a, b)$ is the Delta function with $\delta(a, b) = 1$ when $a = b$, and 0 otherwise. The conditional dependency of $\boldsymbol{W}$, $\boldsymbol{Z}$, and G can be represented as a graphical model [2]. The problem of learning $\boldsymbol{W}$ is equivalent to learning $\boldsymbol{Z}$ and G, that

[2] $\boldsymbol{Z} \rightarrow \boldsymbol{W} \leftarrow G$: $\boldsymbol{Z}$ and G are dependent given $\boldsymbol{W}$.

is learning word distributions among the groups and group importance. According to $\delta(Z_{i,j}, \iota)$ and $\delta(G, \iota)$, the word masks $\boldsymbol{W}$ will keep or mask out all words in group ι, which satisfies the third property.

3.2 Learning GMASK

We formulate the problem of learning GMASK by optimizing the following objective in terms of the three properties,

$$\max_{\Phi,\Psi} \mathbb{E}[p(y \mid \boldsymbol{x}, \boldsymbol{z}, \boldsymbol{g})] - \gamma_1 \mathcal{L}_{\boldsymbol{Z}} - \gamma_2 \mathcal{L}_G, \quad (3)$$

where Φ and Ψ are parameters of $\boldsymbol{Z}$ and G respectively, and $\boldsymbol{z}$ and $\boldsymbol{g}$ are samples of $\boldsymbol{Z}$ and G respectively. We denote $\mathcal{L}_{\boldsymbol{Z}}$ and $\mathcal{L}_G$ as regularizations on $\boldsymbol{Z}$ and G respectively, which are applied to make the learned masks satisfy the required properties. We will introduce the two regularization terms subsequently. $\gamma_1, \gamma_2 \in \mathbb{R}_+$ are coefficients.

Optimizing the first term in Equation 3 is to make the word masks $\boldsymbol{W}$ satisfy the first property, that is the model outputs the same prediction on the selected words as on the whole text. Given $\boldsymbol{Z}$ and G, we have word masks $\boldsymbol{W}$, and multiply them with input word embeddings, and obtain the masked input $\tilde{\boldsymbol{x}}$ as in Equation 1. The model output on $\tilde{\boldsymbol{x}}$ is $\tilde{y} = f(\tilde{\boldsymbol{x}})$. If the masks correctly select important words, the predicted label on the selected words should be equal to that on the whole input text. We can optimize the first term by minimizing the cross entropy loss ($\mathcal{L}_{ce}(\cdot,\cdot)$) between $\tilde{y}$ and y. The objective Equation 3 can be rewritten as

$$\min_{\Phi,\Psi} \mathcal{L}_{ce}(y, \tilde{y}) + \gamma_1 \mathcal{L}_{\boldsymbol{Z}} + \gamma_2 \mathcal{L}_G. \quad (4)$$

The last two terms in the optimization objective are to make word masks satisfy the second and third properties. We regularize $\boldsymbol{Z}$ to encourage each group contains some words from different sentences. We regularize G to ensure only one or few groups are identified as important (with relatively large probabilities). Optimizing the cross entropy loss with the two regularization terms can make the word masks select the important group of words, where the words are selected from the input sentence pair and are correlated.

Regularizations on $\boldsymbol{Z}$ and G As each $Z_{i,j}$ ($i \in \{1,2\}$, $j \in \{1,\ldots,n_i\}$) indicates a word belonging to a specific group, it follows categorical distribution with probabilities $[\phi_{i,j}(1), \ldots, \phi_{i,j}(t)]$, where t is the predefined number of groups, and $\phi_{i,j}(\iota)$ ($\iota \in \{1,\ldots,t\}$) represents the probability of the word in group ι. Then we denote the parameters of $\boldsymbol{Z}$ as Φ,

$$\Phi = \begin{bmatrix} \phi_{1,1}(1) & \cdots & \phi_{1,1}(t) \\ \vdots & \cdots & \vdots \\ \phi_{1,n_1}(1) & \cdots & \phi_{1,n_1}(t) \\ \phi_{2,1}(1) & \cdots & \phi_{2,1}(t) \\ \vdots & \cdots & \vdots \\ \phi_{2,n_2}(1) & \cdots & \phi_{2,n_2}(t) \end{bmatrix} \quad (5)$$

To ensure that each group contains some words from both input sentences, and also avoid assigning a bunch of words into one group, we distribute the words in each sentence evenly among all groups. Then each group implicitly captures the words from different sentences. We can regularize $\boldsymbol{Z}$ to achieve this goal. We sum each column of Φ along the upper half rows and lower half rows respectively, and obtain two vectors by taking averages, $\boldsymbol{\phi}^U = \frac{1}{n_1}[\sum_{j=1}^{n_1} \phi_{1,j}(1), \ldots, \sum_{j=1}^{n_1} \phi_{1,j}(t)]$, $\boldsymbol{\phi}^L = \frac{1}{n_2}[\sum_{j=1}^{n_2} \phi_{2,j}(1), \ldots, \sum_{j=1}^{n_2} \phi_{2,j}(t)]$. Then $\boldsymbol{\phi}^U$ and $\boldsymbol{\phi}^L$ are the distributions of two discrete variables Z^U and Z^L, which also represent the word distributions of the two input sentences among groups. To make the distributions of words even, we maximize the entropy of Z^U and Z^L, and have

$$\mathcal{L}_{\boldsymbol{Z}} = -(H(Z^U) + H(Z^L)), \quad (6)$$

where $H(\cdot)$ is entropy.

$G \in \{1,\ldots,t\}$ also follows categorical distribution with probabilities $\Psi = [\psi(1), \ldots, \psi(t)]$, where $\psi(\iota)$ ($\iota \in \{1,\ldots,t\}$) represents the probability of group ι being selected. According to the relation of $\boldsymbol{W}$, $\boldsymbol{Z}$, G in Equation 2, the word masks only keep the words assigned to the selected group. To ensure one or few groups have relatively large probabilities to be selected, we regularize G by minimizing its entropy, that is $\mathcal{L}_G = H(G)$. The final optimization objective is

$$\min_{\Phi,\Psi} \mathcal{L}_{ce}(y, \tilde{y}) - \gamma_1(H(Z^U) + H(Z^L)) + \gamma_2 H(G). \quad (7)$$

Optimization via sampling We adopt a sampling based method to solve Equation 7 by learning the parameters of $\boldsymbol{Z}$ and G (i.e. $\{\Phi, \Psi\}$). As the right part of Figure 2 shows, we sample a $\boldsymbol{z}$ from the categorical distributions of $\boldsymbol{Z}$, where each row

$\boldsymbol{z}_{i,j}$ is a one-hot vector, indicating the word $\boldsymbol{x}_{i,j}$ assigned to a specific group. And we sample a $\boldsymbol{g}$ from the categorical distribution of G, which is a vertical one-hot vector, indicating the selected group. Then we obtain a sample of word masks by multiplying $\boldsymbol{z}$ and $\boldsymbol{g}$, i.e. $\boldsymbol{w} = \boldsymbol{z} \cdot \boldsymbol{g}$, where the mask values corresponding to the words in the selected group are 1, while the rest are 0. We apply the masks on the input word embeddings and optimize Equation 7 via stochastic gradient descent.

There are two challenges of the learning process: discreteness and large variance. We apply the Gumbel-softmax trick (Jang et al., 2016; Maddison et al., 2016) to address the discreteness of sampling from categorical distributions in backpropagation. See Appendix A for the continuous differentiable approximation of Gumbel-softmax. We do the sampling multiple times in subsection 3.3 and generate a batch of masked inputs of the original input data to decrease the variance in probing the model, and train for multiple epochs until the learnable parameters $\{\Phi, \Psi\}$ reach stable values.

Weighted word attributions After training, we learn the parameters of $\boldsymbol{Z}$, i.e. Φ, where each element $\phi_{i,j}(\iota) \in (0,1)$ ($i \in \{1,2\}$, $j \in \{1,\ldots,n_i\}$, $\iota \in \{1,\ldots,t\}$) represents the probability of word $\boldsymbol{x}_{i,j}$ belonging to group ι. We also learn the parameters of G, i.e. Ψ, where each element $\psi(\iota) \in (0,1)$ represents the importance of group ι. According to the definition of word masks $\boldsymbol{W}$ in subsection 3.1, we know that each mask variable $W_{i,j}$ follows Bernoulli distribution, and the probability of $W_{i,j}$ taking 1 is denoted as $\theta_{i,j}$. We can compute $\theta_{i,j}$ based on the relation of $W_{i,j}$, $Z_{i,j}$ and G in Equation 2, that is

$$\theta_{i,j} = \sum_{\iota=1}^{t} \phi_{i,j}(\iota)\psi(\iota). \tag{8}$$

We can see that $\theta_{i,j}$ is the expectation of $W_{i,j}$, representing the weighted attribution of the word $\boldsymbol{x}_{i,j}$ to the model predicition. Hence, we have a set of weighted word attributions $\Theta = [\theta_{1,1},\ldots,\theta_{1,n_1},\theta_{2,1},\ldots,\theta_{2,n_2}]^T$ for extracting important words as an explanation.

Complexity For a set of n words, computing interactions between all word pairs costs $O(n^2)$ and aggregating words step by step to form a tree structure even costs more time (Singh et al., 2018; Chen et al., 2020). GMASK circumvents the feature interaction detection by learning word groups. The complexity is $O(nt + t)$, where t is the number of groups and usually $t \ll n$ in practice.

3.3 Implementation Specification

We initialize the parameters of all categorical distributions ($\{\Phi, \Psi\}$) with $\frac{1}{t}$, which means all words have the same importance and do not have any preference to be in a specific group at the start of training. To stabilize the learning process, we sample 100 - 1000 examples (depending on the model and datasets) and train at most 100 epochs until converge. The coefficients γ_1 and γ_2 are hyper-parameters. We empirically found $\gamma_1 = 10$ and $\gamma_2 = 1$ work well in our experiments.

In our pilot experiments, we found that preliminarily filtering out some noisy or irrelevant words can help decrease the learnable parameters, hence accelerating the training process. Specifically, we adopt a simple word mask method from (Chen and Ji, 2020) to select a set of individual words for an input sentence pair before running GMASK. This simple method, denoted as IMASK, will learn individual word attributions as masks $\boldsymbol{R} = \{R_{i,j}\}_{i\in\{1,2\},\ j\in\{1,\ldots,n_i\}} \in \{0,1\}^{n_1+n_2}$ regardless any correlation. Then, based on the expected values of $\boldsymbol{R}$, we preliminarily select top k words for GMASK to further learn weighted word attributions. Within these top k words, assume k_1 words from the first input text and k_2 words from the second text, then we will set the number of groups as $t = \min(k_1, k_2)$, so that at least one group contains words from both sentences. k is a hyper-parameter associated with the average length of input texts. In the experiments, we set $k = 10$. Note that, the IMASK method adopted here can also be used as a baseline method for comparison.

4 Experimental Setup

We evaluate GMASK with two kinds of neural network models, decomposable attention model (DAttn) (Parikh et al., 2016) and BERT (Devlin et al., 2018), for two types of sentence pair modeling tasks on four datasets. We compare our method with four baselines.

Datasets e-SNLI (Camburu et al., 2018) is natural language inference task, where the model predicts the semantic relationship between two input sentences as entailment, contradiction, or neutral. Quora (Wang et al., 2017), QQP (Wang et al., 2018) and MRPC (Dolan and Brockett, 2005) are paraphrase identification tasks, where the model

Models	e-SNLI	Quora	QQP	MRPC
DAttn	86.62	86.78	85.00	68.30
BERT	90.38	90.48	89.00	83.70

Table 1: The prediction accuracy (%) of different models on the four datasets.

decides whether two input texts are semantically equivalent or not. The statistics of the four datasets are in Appendix B.

Models We adopt the decomposable attention model (DAttn) (Parikh et al., 2016) and BERT (Devlin et al., 2018) model, and fine-tune the models on each downstream task to achieve the best performance, as Table 1 shows. The test results on QQP and MRPC are scored by the GLUE benchmark [3]. The corresponding validation accuracy for each reported test accuracy is in Appendix C

Baselines We compare GMASK with four baseline methods: (1) LIME (Ribeiro et al., 2016)-fitting a local linear model with perturbations to approximate the neural network and produce word attributions; (2) L2X (Chen et al., 2018)-constructing a network to learn feature attributions by maximizing the mutual information between the selected features and model output; (3) IBA (*Per-Sample* framework) (Schulz et al., 2020) - learning feature attributions by optimizing the information bottleneck which restricts feature information flow by adding noise; (4) IMASK (subsection 3.3)-learning individual word masks. Note that here we use standalone IMASK as one of the baselines, as oppose to applying it for selecting preliminary important words for GMASK as in subsection 3.3.

More details about experimental setup are in Appendix B, including data pre-processing and model configurations.

5 Results and Discussion

We compare the faithfulness of generated post-hoc explanations via both quantitative and qualitative evaluations.

5.1 Quantitative Evaluation

We adopt three metrics from prior work to evaluate the faithfulness of learned feature attributions: AOPC score (Nguyen, 2018; Samek et al., 2016), post-hoc accuracy (Chen et al., 2018; Chen and Ji, 2020), and degradation score (Ancona et al., 2017; Schulz et al., 2020). We evaluate explanations on all test data for the MRPC dataset, and on 2000 examples randomly selected from the test set for other three datasets due to computational complexity. The average runtime is in Appendix D.

[3] https://gluebenchmark.com/

Models	Methods	e-SNLI	Quora	QQP	MRPC
DAttn	LIME	0.286	0.120	0.079	0.064
	L2X	0.299	0.128	0.079	0.035
	IBA	0.354	0.137	**0.104**	**0.109**
	IMASK	0.324	0.140	0.087	0.064
	GMASK	**0.361**	**0.142**	0.095	0.091
BERT	LIME	0.221	0.153	0.110	0.062
	L2X	0.310	0.119	0.134	0.083
	IBA	0.282	0.199	0.144	0.114
	IMASK	0.292	0.232	0.139	0.130
	GMASK	**0.319**	**0.309**	**0.181**	**0.200**

Table 2: AOPC scores of different explanation methods with the DAttn and BERT models on the four datasets.

5.1.1 AOPC score

We adopt the area over the perturbation curve (AOPC) (Nguyen, 2018; Samek et al., 2016) metric to evaluate the comprehensiveness of explanations to models. It calculates the average change of prediction probability on the predicted class over all examples by removing top $1 \dots u$ words in explanations.

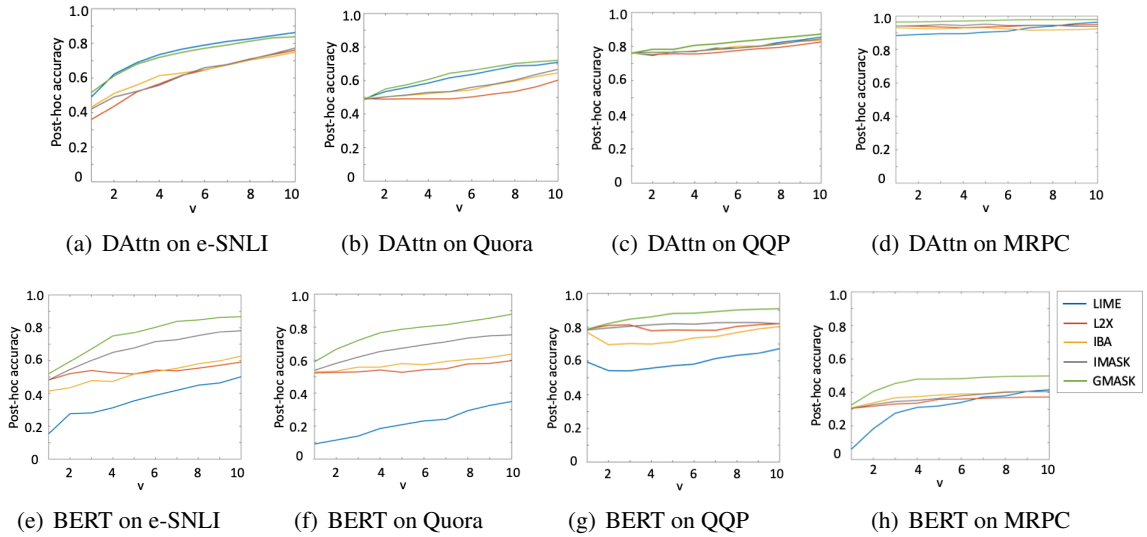
$$\text{AOPC} = \frac{1}{U+1}\langle \sum_{u=1}^{U} p(y|\boldsymbol{x}) - p(y|\boldsymbol{x}_{\setminus 1 \dots u}) \rangle_{\boldsymbol{x}}, \tag{9}$$

where $p(y|\boldsymbol{x}_{\setminus 1 \dots u})$ is the probability for the predicted class when words $1 \dots u$ are removed and $\langle \cdot \rangle_{\boldsymbol{x}}$ denotes the average over all test examples. Higher AOPC score indicates better explanations.

Table 2 shows the results of AOPC scores of different explanation methods when $U = 10$. GMASK outperforms other baseline methods on most of the datasets. Especially for the BERT model, GMASK achieves significantly higher AOPC scores than other methods, indicating that BERT tends to rely on word correlations to make predictions. IBA and IMASK, either learning continuous or binary individual word masks, perform better than learning word attributions via an additional network (L2X) or using linear approximation (LIME).

5.1.2 Post-hoc Accuracy

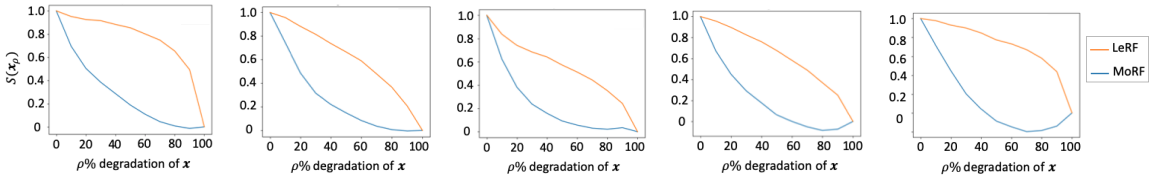
The post-hoc accuracy (Chen et al., 2018; Chen and Ji, 2020) evaluates the sufficiency of important



(a) DAttn on e-SNLI (b) DAttn on Quora (c) DAttn on QQP (d) DAttn on MRPC

(e) BERT on e-SNLI (f) BERT on Quora (g) BERT on QQP (h) BERT on MRPC

Figure 3: Post-hoc accuracy of different explanation methods with the DAttn and BERT models on the four datasets.



(a) LIME (b) L2X (c) IBA (d) IMASK (e) GMASK

Figure 4: Degradation test of different explanation methods with the DAttn model on the e-SNLI dataset.

words to the model prediction. For each test data, we select top v important words based on word attributions for the model to make a prediction, and compare it with the original prediction made on the whole input text. We compute the post-hoc accuracy on M examples,

$$\text{post-hoc-acc}(v) = \frac{1}{M}\sum_{m=1}^{M} \mathbb{1}[y_v^{(m)} = y^{(m)}],$$

where $y^{(m)}$ is the predicted label on the m-th test data, and $y_v^{(m)}$ is the predicted label based on the top v important words. Higher post-hoc accuracy indicates better explanations.

Figure 3 shows the results of post-hoc accuracy of different explanation methods where we increase v from 1 to 10. Similar to the results of the AOPC scores, GMASK achieves higher post-hoc accuracy than other methods for both DAttn and BERT models.

The explanations of GMASK for the BERT model achieve about 80% post-hoc accuracy on all datasets except the MRPC dataset. This is only by relying on top 4 important words, which means that GMASK captures informative words for model predictions. The post-hoc accuracies of the BERT model on the MRPC dataset are lower than those on other three datasets because the average sentence length of MRPC is twice as long as the others, indicating that BERT tends to use larger context for predictions. The post-hoc accuracies of the DAttn model on the MRPC dataset are extremely high for all the explanation methods. The reason is that the prediction accuracy of DAttn model on the MRPC dataset is relatively low (Table 1). Any random words picked up by explanations could make the model output wrong predictions since the original predictions on the whole texts are also wrong, hence causing high post-hoc accuracy.

5.1.3 Degradation Test

Degradation test (Ancona et al., 2017; Schulz et al., 2020) evaluates the ranking of importance by removing the most important words or least important words first, and observing model prediction probability drop on the predicted class. We draw

Models	Methods	Texts
DAttn	LIME	a man playing an electric guitar on stage . a man playing banjo on the floor .
	L2X	a man playing an electric guitar on stage . a man playing banjo on the floor .
	IBA	a man playing an electric guitar on stage . a man playing banjo on the floor .
	IMASK	a man playing an electric guitar on stage . a man playing banjo on the floor .
	GMASK	a man playing an electric guitar on stage . a man playing banjo on the floor .
BERT	LIME	why are vikings portrayed wearing horned helmets ? why did vikings have horns on their helmets ?
	L2X	why are vikings portrayed wearing horned helmets ? why did vikings have horns on their helmets ?
	IBA	why are vikings portrayed wearing horned helmets ? why did vikings have horns on their helmets ?
	IMASK	why are vikings portrayed wearing horned helmets ? why did vikings have horns on their helmets ?
	GMASK	why are vikings portrayed wearing horned helmets ? why did vikings have horns on their helmets ?

Table 3: Examples of different explanations, where the top four important words are highlighted. The important words in the first and second sentences are highlighted in pink and blue colors respectively. The color saturation indicates word attribution. The first example is from the e-SNLI dataset, and the DAttn model makes a correct prediction as CONTRADICTION. The second example is from the Quora dataset, and the BERT model makes a correct prediction as PARAPHRASES.

Models	Methods	e-SNLI	Quora	QQP	MRPC
DAttn	LIME	0.502	0.070	0.091	1.367
	L2X	0.366	0.002	0.036	1.779
	IBA	0.423	0.110	0.197	2.775
	IMASK	0.436	0.152	0.214	2.037
	GMASK	**0.620**	**0.178**	**0.238**	**2.790**
BERT	LIME	0.188	0.192	0.087	0.018
	L2X	0.303	0.168	0.173	-0.003
	IBA	0.166	0.038	0.176	0.050
	IMASK	0.369	0.303	0.172	0.251
	GMASK	**0.576**	**0.726**	**0.707**	**0.533**

Table 4: Degradation scores of different explanation methods with the DAttn and BERT models on the four datasets.

two curves as shown in Figure 4, one with the most relevant words removed first (MoRF) and another one with the least relevant words removed first (LeRF). x-axis is the percentage of words removed (degradation proportion), and y-axis is the normalized model output probability as

$$S(\boldsymbol{x}_\rho) = \frac{p(y|\boldsymbol{x}_\rho) - p(y|\boldsymbol{x}_o)}{p(y|\boldsymbol{x}) - p(y|\boldsymbol{x}_o)}, \quad (10)$$

where $\boldsymbol{x}$ is the original input, y is the predicted label, $\boldsymbol{x}_\rho$ means $\rho\%(\rho \in [0, 100])$ degradation of $\boldsymbol{x}$, and $\boldsymbol{x}_o$ is full degradation. We compute the averages of $p(y|\boldsymbol{x}_\rho)$, $p(y|\boldsymbol{x})$, and $p(y|\boldsymbol{x}_o)$ over all test examples. The degradation score is calculated as the integral between the MoRF and LeRF curves,

$$\text{degra-score} = \int_{\rho=0}^{100} \frac{S^L(\boldsymbol{x}_\rho) - S^M(\boldsymbol{x}_\rho)}{100} d\rho, \quad (11)$$

where $S^L(\boldsymbol{x}_\rho)$ and $S^M(\boldsymbol{x}_\rho)$ are normalized model outputs by removing the least or most important words respectively. Higher degradation score is better.

Table 4 shows the results of degradation scores of different explanation methods. GMASK shows superiority to other baseline methods under this metric. Figure 4 shows the degradation test results of DAttn model on the e-SNLI dataset. GMASK can distinguish both important and unimportant words, while IBA does not learn the correct order of unimportant words. LIME does not perform well in identifying important words, but captures the correct order of unimportant words. The MoRF and LeRF curves of L2X and IMASK are relatively symmetric, but not as expanded as GMASK.

5.2 Qualitative Evaluation

Table 3 shows different explanations on two examples from e-SNLI and Quora respectively. See Appendix E for more examples. For the first example, the DAttn model makes a correct prediction as CONTRADICTION. For the second example, the BERT model also makes a correct prediction as PARAPHRASES. We highlight the top four important words, where the words in the first and second sentences are in pink and blue colors respectively. The color saturation indicates word attribution.

For the first example, LIME and IBA mainly capture the important words from the first sentence, while ignoring the ones in the second sentence (e.g. `banjo`, `floor`). On the contrary, L2X focuses on the words in the second sentence, while ignoring the important word `guitar` in the first sentence. IMASK picks up two irrelevant words `man` and `a` as important words, which can not explain the model prediction. GMASK correctly identifies top

four important words and captures two correlated words `guitar` and `banjo` from the two input sentences respectively.

For the second example, only GMASK captures the two important correlated words `horned` and `horns`, which explains why the BERT model predicts the two input questions as paraphrases. LIME captures the overlapped word `helmets` in the two sentences, while L2X only selects some irrelevant words. Both IBA and IMASK identify a question mark as the important word, which is untrustworthy to the model prediction.

6 Conclusion

In this paper, we focused on sentence pair modeling and proposed an effective method, GMASK, learning group masks for correlated words and calculating weighted word attributions. We tested GMASK with two different neural network models on four datasets, and assessed its effectiveness via both quantitative and qualitative evaluations.

7 Ethical Considerations

The motivation of this work is aligned with the merits of explainable AI, in which the goal is to increase the trustworthiness of neural network models in decision making. One potential ethical concern of this work is that explanations can be used to design adversarial examples for attacking. Although the main focus of this work is about generating faithful explanations, we do realize the importance of whether human users can actually understand explanations. To address this concern, a better strategy is to collaborate with HCI experts in our future work. In addition, we provide necessary implementation details to make sure the results in this paper are reproducible.

References

Marco Ancona, Enea Ceolini, Cengiz Öztireli, and Markus Gross. 2017. A unified view of gradient-based attribution methods for deep neural networks. In *NIPS Workshop on Interpreting, Explaining and Visualizing Deep Learning*.

Jasmijn Bastings, Wilker Aziz, and Ivan Titov. 2019. Interpretable neural predictions with differentiable binary variables. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2963–2977, Florence, Italy. Association for Computational Linguistics.

Piotr Bojanowski, Edouard Grave, Armand Joulin, and Tomas Mikolov. 2017. Enriching word vectors with subword information. *Transactions of the Association for Computational Linguistics*, 5:135–146.

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*, pages 632–642, Lisbon, Portugal. Association for Computational Linguistics.

Oana-Maria Camburu, Tim Rocktäschel, Thomas Lukasiewicz, and Phil Blunsom. 2018. e-snli: Natural language inference with natural language explanations. In *Advances in Neural Information Processing Systems*, pages 9539–9549.

Hanjie Chen and Yangfeng Ji. 2020. Learning variational word masks to improve the interpretability of neural text classifiers. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 4236–4251, Online. Association for Computational Linguistics.

Hanjie Chen, Guangtao Zheng, and Yangfeng Ji. 2020. Generating hierarchical explanations on text classification via feature interaction detection. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5578–5593, Online. Association for Computational Linguistics.

Jianbo Chen, Le Song, Martin Wainwright, and Michael Jordan. 2018. Learning to explain: An information-theoretic perspective on model interpretation. In *International Conference on Machine Learning*, pages 883–892. PMLR.

Nicola De Cao, Michael Sejr Schlichtkrull, Wilker Aziz, and Ivan Titov. 2020. How do decisions emerge across layers in neural models? interpretation with differentiable masking. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 3243–3255, Online. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*.

William B Dolan and Chris Brockett. 2005. Automatically constructing a corpus of sentential paraphrases. In *Proceedings of the Third International Workshop on Paraphrasing (IWP2005)*.

Mengnan Du, Ninghao Liu, Fan Yang, Shuiwang Ji, and Xia Hu. 2019. On attribution of recurrent neural network predictions via additive decomposition. In *The World Wide Web Conference*, pages 383–393.

Reza Ghaeini, Xiaoli Z Fern, and Prasad Tadepalli. 2018. Interpreting recurrent and attention-based neural models: a case study on natural language inference. *arXiv preprint arXiv:1808.03894*.

Riccardo Guidotti, Anna Monreale, Salvatore Ruggieri, Franco Turini, Fosca Giannotti, and Dino Pedreschi. 2018. A survey of methods for explaining black box models. *ACM computing surveys (CSUR)*, 51(5):1–42.

Yotam Hechtlinger. 2016. Interpretation of prediction models using the input gradient. *arXiv preprint arXiv:1611.07634*.

Sarthak Jain and Byron C Wallace. 2019. Attention is not explanation. *arXiv preprint arXiv:1902.10186*.

Eric Jang, Shixiang Gu, and Ben Poole. 2016. Categorical reparameterization with gumbel-softmax. *arXiv preprint arXiv:1611.01144*.

Xisen Jin, Junyi Du, Zhongyu Wei, Xiangyang Xue, and Xiang Ren. 2019. Towards hierarchical importance attribution: Explaining compositional semantics for neural sequence models. *arXiv preprint arXiv:1911.06194*.

Tao Lei, Regina Barzilay, and Tommi Jaakkola. 2016. Rationalizing neural predictions. *arXiv preprint arXiv:1606.04155*.

Jiwei Li, Will Monroe, and Dan Jurafsky. 2016. Understanding neural networks through representation erasure. *arXiv preprint arXiv:1612.08220*.

Scott M Lundberg, Gabriel G Erion, and Su-In Lee. 2018. Consistent individualized feature attribution for tree ensembles. *arXiv preprint arXiv:1802.03888*.

Scott M Lundberg and Su-In Lee. 2017. A unified approach to interpreting model predictions. In *Advances in neural information processing systems*, pages 4765–4774.

Chris J Maddison, Andriy Mnih, and Yee Whye Teh. 2016. The concrete distribution: A continuous relaxation of discrete random variables. *arXiv preprint arXiv:1611.00712*.

W James Murdoch, Peter J Liu, and Bin Yu. 2018. Beyond word importance: Contextual decomposition to extract interactions from lstms. *arXiv preprint arXiv:1801.05453*.

Dong Nguyen. 2018. Comparing automatic and human evaluation of local explanations for text classification. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1069–1078.

Ankur P Parikh, Oscar Täckström, Dipanjan Das, and Jakob Uszkoreit. 2016. A decomposable attention model for natural language inference. *arXiv preprint arXiv:1606.01933*.

Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. 2016. " why should i trust you?" explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pages 1135–1144.

Tim Rocktäschel, Edward Grefenstette, Karl Moritz Hermann, Tomáš Kočiskỳ, and Phil Blunsom. 2015. Reasoning about entailment with neural attention. *arXiv preprint arXiv:1509.06664*.

Wojciech Samek, Alexander Binder, Grégoire Montavon, Sebastian Lapuschkin, and Klaus-Robert Müller. 2016. Evaluating the visualization of what a deep neural network has learned. *IEEE transactions on neural networks and learning systems*, 28(11):2660–2673.

Karl Schulz, Leon Sixt, Federico Tombari, and Tim Landgraf. 2020. Restricting the flow: Information bottlenecks for attribution. *arXiv preprint arXiv:2001.00396*.

Sofia Serrano and Noah A Smith. 2019. Is attention interpretable? *arXiv preprint arXiv:1906.03731*.

Lloyd S Shapley. 1953. A value for n-person games. *Contributions to the Theory of Games*, 2(28).

Chandan Singh, W James Murdoch, and Bin Yu. 2018. Hierarchical interpretations for neural network predictions. *arXiv preprint arXiv:1806.05337*.

Mukund Sundararajan, Ankur Taly, and Qiqi Yan. 2017. Axiomatic attribution for deep networks. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 3319–3328. JMLR. org.

James Thorne, Andreas Vlachos, Christos Christodoulopoulos, and Arpit Mittal. 2019. Generating token-level explanations for natural language inference. *arXiv preprint arXiv:1904.10717*.

Naftali Tishby, Fernando C Pereira, and William Bialek. 2000. The information bottleneck method. *arXiv preprint physics/0004057*.

Michael Tsang, Dehua Cheng, Hanpeng Liu, Xue Feng, Eric Zhou, and Yan Liu. 2020. Feature interaction interpretability: A case for explaining ad-recommendation systems via neural interaction detection. *arXiv preprint arXiv:2006.10966*.

Michael Tsang, Youbang Sun, Dongxu Ren, and Yan Liu. 2018. Can i trust you more? model-agnostic hierarchical explanations. *arXiv preprint arXiv:1812.04801*.

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. *arXiv preprint arXiv:1804.07461*.

Zhiguo Wang, Wael Hamza, and Radu Florian. 2017. Bilateral multi-perspective matching for natural language sentences. *arXiv preprint arXiv:1702.03814*.

Sarah Wiegreffe and Yuval Pinter. 2019. Attention is not not explanation. *arXiv preprint arXiv:1908.04626*.

Wenpeng Yin and Hinrich Schütze. 2015. Convolutional neural network for paraphrase identification. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 901–911.

A Sampling with Gumbel-softmax Trick

For a random variable $A \in \{1, \ldots, t\}$ which follows categorical distribution with probabilities $[\lambda_1, \ldots, \lambda_t]$. We draw samples from a Gumbel(0, 1) distribution for each category $\iota \in \{1, \ldots, t\}$:

$$s_\iota = -\log(-\log u),\ u \sim \text{Uniform}(0, 1), \quad (12)$$

and then apply a temperature-dependent softmax over the t categories with each dimension calculated as

$$a_{s_\iota} = \frac{\exp((\log(\lambda_\iota) + s_\iota)/\tau)}{\sum_\iota \exp((\log(\lambda_\iota) + s_\iota)/\tau)}, \quad (13)$$

where τ is a hyperparameter called the softmax temperature.

B Supplement of Experiment Setup

Datasets The datasets are all in English. Table 5 shows the statistics of the four datasets. We adopt the data splits of e-SNLI (Camburu et al., 2018) from the ERASER benchmark [4]. We adopt the data splits of Quora released by Wang et al. (2017). The data splits of QQP (Wang et al., 2018) and MRPC (Dolan and Brockett, 2005) are from the GLUE benchmark. We clean up the text by converting all characters to lowercase, removing extra whitespaces and special characters, and build vocabulary.

Models We set the hidden size of feed forward networks in the DAttn model (Parikh et al., 2016) as 300, and initialize word embeddings with pre-trained `fastText` (Bojanowski et al., 2017). For BERT model (Devlin et al., 2018), we use the pre-trained BERT-base model[5] with 12 transformer layers, 12 self-attention heads, and the hidden size of 768.

We implement the models in PyTorch 3.6. The number of parameters in the DAttn and BERT models are 11046303 and 109484547 respectively. We fine-tune hyperparameters manually for each model to achieve the best prediction accuracy, such as learning rate $lr \in \{1e-4, 1e-3, \cdots, 1\}$, clipping norm $clip \in \{1e-3, 1e-2, \cdots, 1, 5, 10\}$.

C Validation Performance

The corresponding validation accuracy for each reported test accuracy is in Table 6.

[4] https://www.eraserbenchmark.com/

[5] https://github.com/huggingface/pytorch-transformers

D Average Runtime

The average runtime of each approach for each model on each dataset is recorded in Table 7. All experiments were performed on a single NVidia GTX 1080 GPU. Note that L2X is efficient in generating explanations for test data, but it costs more time on training the interpretation model on the whole training set.

E Examples of Different Explanations

Table 8 shows more examples of different explanations for the DAttn and BERT model on different datasets.

Datasets	*C*	*L*	*V*	*#train*	*#dev*	*#test*
e-SNLI	3	10.2	64291	549K	9K	9K
Quora	2	11.5	85249	384K	10K	10K
QQP	2	11.1	126266	364K	40K	391K
MRPC	2	22	15547	3668	408	1725

Table 5: Summary statistics for the datasets, where *C* is the number of classes, *L* is average sentence length, *V* is vocab size, and # counts the number of examples in the *train/dev/test* sets.

Models	e-SNLI	Quora	QQP	MRPC
DAttn	87.75	87.36	87.19	73.77
BERT	90.43	91.21	91.31	86.52

Table 6: The validation accuracy (%) of different models on the four datasets.

Models	Methods	e-SNLI	Quora	QQP	MRPC
DAttn	LIME	20.17	20.45	20.43	19.12
	L2X	0.01	0.02	0.02	0.01
	IBA	9.87	9.93	9.96	9.24
	IMASK	0.17	0.19	0.18	0.12
	GMASK	11.10	11.23	11.25	10.96
BERT	LIME	13.41	14.01	14.08	13.21
	L2X	0.02	0.02	0.02	0.01
	IBA	3.29	3.38	3.34	3.20
	IMASK	0.62	0.70	0.71	0.59
	GMASK	43.24	43.56	43.77	42.84

Table 7: The average runtime (s/example) of each approach for each model on each dataset.

Model/Dataset/Prediction	Methods	Texts
DAttn/Quora/PARAPHRASES	LIME	who are some famous nihilists ? what would a nihilistic president do to the us ?
	L2X	who are some famous nihilists ? what would a nihilistic president do to the us ?
	IBA	who are some famous nihilists ? what would a nihilistic president do to the us ?
	IMASK	who are some famous nihilists ? what would a nihilistic president do to the us ?
	GMASK	who are some famous nihilists ? what would a nihilistic president do to the us ?
DAttn/QQP/NONPARAPHRASES	LIME	can i register shares of a private limited company in india ? can a school in india be registered as a private limited company ?
	L2X	can i register shares of a private limited company in india ? can a school in india be registered as a private limited company ?
	IBA	can i register shares of a private limited company in india ? can a school in india be registered as a private limited company ?
	IMASK	can i register shares of a private limited company in india ? can a school in india be registered as a private limited company ?
	GMASK	can i register shares of a private limited company in india ? can a school in india be registered as a private limited company ?
DAttn/MRPC/PARAPHRASES	LIME	these documents are indecipherable to me and the fact is that this investigation has led nowhere the lawyer said . these documents are indecipherable to me the lawyers said and the fact is that this investigation has led nowhere .
	L2X	these documents are indecipherable to me and the fact is that this investigation has led nowhere the lawyer said . these documents are indecipherable to me the lawyers said and the fact is that this investigation has led nowhere .
	IBA	these documents are indecipherable to me and the fact is that this investigation has led nowhere the lawyer said . these documents are indecipherable to me the lawyers said and the fact is that this investigation has led nowhere .
	IMASK	these documents are indecipherable to me and the fact is that this investigation has led nowhere the lawyer said . these documents are indecipherable to me the lawyers said and the fact is that this investigation has led nowhere .
	GMASK	these documents are indecipherable to me and the fact is that this investigation has led nowhere the lawyer said . these documents are indecipherable to me the lawyers said and the fact is that this investigation has led nowhere .
BERT/e-SNLI/CONTRADICTION	LIME	a band singing and playing electric guitar for a crowd of people . the band is backstage .
	L2X	a band singing and playing electric guitar for a crowd of people . the band is backstage .
	IBA	a band singing and playing electric guitar for a crowd of people . the band is backstage .
	IMASK	a band singing and playing electric guitar for a crowd of people . the band is backstage .
	GMASK	a band singing and playing electric guitar for a crowd of people . the band is backstage .
BERT/QQP/PARAPHRASES	LIME	how do i quit smoking ? how do i give up on cigarette smoking ?
	L2X	how do i quit smoking ? how do i give up on cigarette smoking ?
	IBA	how do i quit smoking ? how do i give up on cigarette smoking ?
	IMASK	how do i quit smoking ? how do i give up on cigarette smoking ?
	GMASK	how do i quit smoking ? how do i give up on cigarette smoking ?
BERT/MRPC/NONPARAPHRASES	LIME	mgm , nbc and liberty executives were not immediately available for comment . a microsoft spokesman was not immediately available to comment .
	L2X	mgm , nbc and liberty executives were not immediately available for comment . a microsoft spokesman was not immediately available to comment .
	IBA	mgm , nbc and liberty executives were not immediately available for comment . a microsoft spokesman was not immediately available to comment .
	IMASK	mgm , nbc and liberty executives were not immediately available for comment . a microsoft spokesman was not immediately available to comment .
	GMASK	mgm , nbc and liberty executives were not immediately available for comment . a microsoft spokesman was not immediately available to comment .

Table 8: Examples of different explanations for the DAttn and BERT model on different datasets.