SelfExplain: A Self-Explaining Architecture for Neural Text Classifiers

Dheeraj Rajagopal♣ **Vidhisha Balachandran**♣ **Eduard Hovy**♣ **Yulia Tsvetkov**♠
♣Language Technologies Institute, Carnegie Mellon University
♠Paul G. Allen School of Computer Science & Engineering, University of Washington
{dheeraj,vbalacha,hovy}@cs.cmu.edu, yuliats@cs.washington.edu

Abstract

We introduce **SelfExplain**, a novel self-explaining model that explains a text classifier's predictions using phrase-based concepts. SelfExplain augments existing neural classifiers by adding (1) a *globally interpretable layer* that identifies the most influential concepts in the training set for a given sample and (2) a *locally interpretable layer* that quantifies the contribution of each local input concept by computing a relevance score relative to the predicted label. Experiments across five text-classification datasets show that SelfExplain facilitates interpretability without sacrificing performance. Most importantly, explanations from SelfExplain show sufficiency for model predictions and are perceived as adequate, trustworthy and understandable by human judges compared to existing widely-used baselines.[1]

1 Introduction

Neural network models are often opaque: they provide limited insight into interpretations of model decisions and are typically treated as "black boxes" (Lipton, 2018). There has been ample evidence that such models overfit to spurious artifacts (Gururangan et al., 2018; McCoy et al., 2019; Kumar et al., 2019) and amplify biases in data (Zhao et al., 2017; Sun et al., 2019). This underscores the need to understand model decision making.

Prior work in interpretability for neural text classification predominantly follows two approaches: (i) *post-hoc explanation methods* that explain predictions for previously trained models based on model internals, and (ii) *inherently interpretable models* whose interpretability is built-in and optimized jointly with the end task. While post-hoc methods (Simonyan et al., 2014; Koh and Liang, 2017; Ribeiro et al., 2016) are often the only option for already-trained models, inherently interpretable models (Melis and Jaakkola, 2018; Arik and Pfister, 2020) may provide greater transparency since explanation capability is embedded directly within the model (Kim et al., 2014; Doshi-Velez and Kim, 2017; Rudin, 2019).

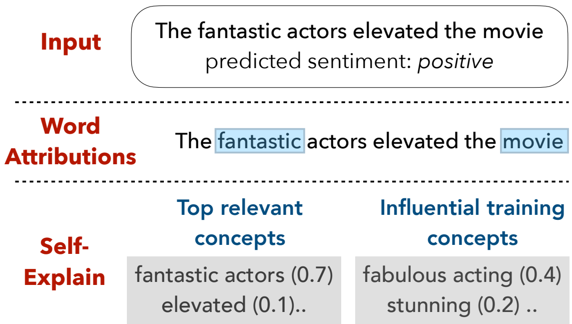

Figure 1: A sample of interpretable concepts from SelfExplain for a binary sentiment analysis task. Compared to saliency-map style word attributions, SelfExplain can provide explanations via concepts in the input sample and the concepts in the training data

In natural language applications, feature attribution based on attention scores (Xu et al., 2015) has been the predominant method for developing inherently interpretable neural classifiers. Such methods interpret model decisions *locally* by explaining the classifier's decision as a function of relevance of features (words) in input samples. However, such interpretations were shown to be unreliable (Serrano and Smith, 2019; Pruthi et al., 2020) and unfaithful (Jain and Wallace, 2019; Wiegreffe and Pinter, 2019). Moreover, with natural language being structured and compositional, explaining the role of higher-level compositional concepts like phrasal structures (beyond individual word-level feature attributions) remains an open challenge. Another known limitation of such feature attribution based methods is that the explanations are limited to the input feature space and often require additional methods (e.g. Han et al., 2020) for providing global

[1]Code and data is publicly available at https://github.com/dheerajrajagopal/SelfExplain

Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, pages 836–850
November 7–11, 2021.

explanations, i.e., explaining model decisions as a function of influential training data.

In this work, we propose SELFEXPLAIN—a self explaining model that incorporates both global and local interpretability layers into neural text classifiers. Compared to word-level feature attributions, we use high-level phrase-based concepts, producing a more holistic picture of a classifier's decisions. SELFEXPLAIN incorporates: (i) *Locally Interpretable Layer* (`LIL`), a layer that quantifies via activation difference, the relevance of each concept to the final label distribution of an input sample. (ii) *Globally Interpretable Layer* (`GIL`), a layer that uses maximum inner product search (MIPS) to retrieve the most influential concepts from the training data for a given input sample. We show how `GIL` and `LIL` layers can be integrated into transformer-based classifiers, converting them into self-explaining architectures. The interpretability of the classifier is enforced through regularization (Melis and Jaakkola, 2018), and the entire model is end-to-end differentiable. To the best of our knowledge, SELFEXPLAIN is the first self-explaining neural text classification approach to provide both global and local interpretability in a single model.

Ultimately, this work makes a step towards combining the generalization power of neural networks with the benefits of interpretable statistical classifiers with hand-engineered features: our experiments on three text classification tasks spanning five datasets with pretrained transformer models show that incorporating `LIL` and `GIL` layers facilitates richer interpretability while maintaining end-task performance. The explanations from SELFEXPLAIN sufficiency reflect model predictions and are perceived by human annotators as more understandable, adequately justifying the model predictions and trustworthy, compared to strong baseline interpretability methods.

2 SELFEXPLAIN

Let $\mathcal{M}$ be a neural C-class classification model that maps $\mathcal{X} \rightarrow \mathcal{Y}$, where $\mathcal{X}$ are the inputs and $\mathcal{Y}$ are the outputs. SELFEXPLAIN builds into $\mathcal{M}$, and it provides a set of explanations $\mathcal{Z}$ via high-level "concepts" that explain the classifier's predictions. We first define interpretable concepts in §2.1. We then describe how these concepts are incorporated into a concept-aware encoder in §2.2. In §2.3, we define our Local Interpretability Layer (`LIL`), which provides local explanations by assigning relevance scores to the constituent concepts of the input. In §2.4, we define our Global Interpretability Layer (`GIL`), which provides global explanations by retrieving influential concepts from the training data. Finally, in §2.5, we describe the end-to-end training procedure and optimization objectives.

2.1 Defining human-interpretable concepts

Since natural language is highly compositional (Montague, 1970), it is essential that interpreting a text sequence goes beyond individual words. We define the set of basic units that are interpretable by humans as *concepts*. In principle, concepts can be words, phrases, sentences, paragraphs or abstract entities. In this work, we focus on phrases as our concepts, specifically all non-terminals in a constituency parse tree. Given any sequence $\mathbf{x} = \{w_i\}_{1:T}$, we decompose the sequence into its component non-terminals $N(\mathbf{x}) = \{nt_j\}_{1:J}$, where J denotes the number of non-terminal phrases in $\mathbf{x}$.

Given an input sample $\mathbf{x}$, $\mathcal{M}$ is trained to produce two types of explanations: (i) global explanations from the training data $\mathcal{X}_{train}$ and (ii) local explanations, which are phrases in $\mathbf{x}$. We show an example in Figure 1. Global explanations are achieved by identifying the most influential concepts $\mathcal{C}_G$ from the "concept store" $\mathbf{Q}$, which is constructed to contain all concepts from the training set $\mathcal{X}_{train}$ by extracting phrases under each non-terminal in a syntax tree for every data sample (detailed in §2.4). Local interpretability is achieved by decomposing the input sample $\mathbf{x}$ into its constituent phrases under each non-terminal in its syntax tree. Then each concept is assigned a score that quantifies its contribution to the sample's label distribution for a given task; $\mathcal{M}$ then outputs the most relevant local concepts $\mathcal{C}_L$.

2.2 Concept-Aware Encoder E

We obtain the encoded representation of our input sequence $\mathbf{x} = \{w_i\}_{1:T}$ from a pretrained transformer model (Vaswani et al., 2017; Liu et al., 2019; Yang et al., 2019) by extracting the final layer output as $\{\mathbf{h}_i\}_{1:T}$. Additionally, we compute representations of concepts, $\{\mathbf{u}_j\}_{1:J}$. For each non-terminal nt_j in $\mathbf{x}$, we represent it as the mean of its constituent word representations $\mathbf{u}_j = \frac{\sum_{w_i \in nt_j} \mathbf{h}_i}{len(nt_j)}$ where $len(nt_j)$ represents the number of words in the phrase nt_j. To represent the root node ($\mathbb{S}$) of the syntax tree, $nt_{\mathbb{S}}$, we use the pooled representation (`[CLS]` token rep-

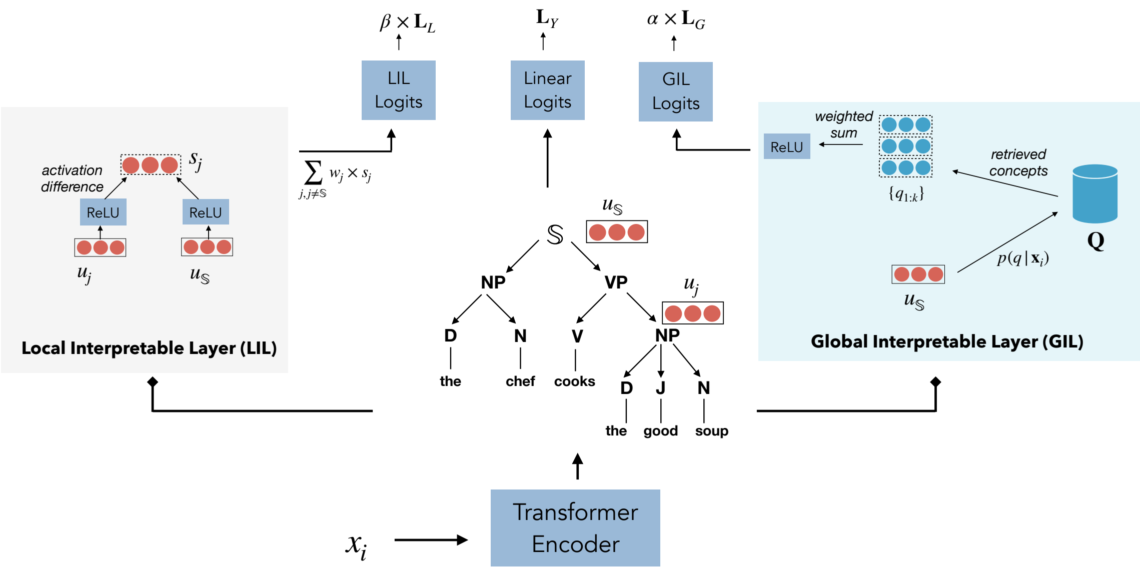

Figure 2: Model Architecture: Our architecture comprises a base encoder that encodes the input and its relative non-terminals. GIL then uses MIPS to retrieve the most influential concepts that *globally* explain the sample, while LIL computes a relevance score for each nt_j that quantifies its relevance to predict the label. The model interpretability is enforced through regularization. Examples of top LIL concepts (extracted from the from input) are {*the good soup*, *good*}, and of top GIL concepts (from the training data) are {*great food, excellent taste*}

resentation) of the pretrained transformer as $\mathbf{u}_{\mathbb{S}}$ for brevity.[2] Following traditional neural classifier setup, the output of the classification layer l_Y is computed as follows:

$$l_Y = \mathtt{softmax}(\mathbf{W}_y \times g(\mathbf{u}_{\mathbb{S}}) + \mathbf{b}_y)$$
$$P_C = \arg\max(l_Y)$$

where g is a $relu$ activation layer, $\mathbf{W}_y \in \mathbb{R}^{D \times C}$, and P_C denotes the index of the predicted class.

2.3 Local Interpretability Layer (LIL)

For local interpretability, we compute a local relevance score for all input concepts $\{nt_j\}_{1:J}$ from the sample $\mathbf{x}$. Approaches that assign relative importance scores to input features through activation differences (Shrikumar et al., 2017; Montavon et al., 2017) are widely adopted for interpretability in computer vision applications. Motivated by this, we adopt a similar approach to NLP applications where we learn the attribution of each concept to the final label distribution via their activation differences. Each non-terminal nt_j is assigned a score that quantifies the contribution of each nt_j to the label in comparison to the contribution of the root node $nt_{\mathbb{S}}$. The most contributing phrases $\mathcal{C}_L$ is used to locally explain the model decisions.

Given the encoder $\mathbf{E}$, LIL computes the contribution solely from nt_j to the final prediction. We first build a representation of the input without contribution of phrase nt_j and use it to score the labels:

$$t_j = g(\mathbf{u}_j) - g(\mathbf{u}_{\mathbb{S}})$$
$$s_j = \mathtt{softmax}(\mathbf{W}_v \times t_j + \mathbf{b}_v)$$

where g is a $relu$ activation function, $t_j \in \mathbb{R}^D$, $s_j \in \mathbb{R}^C$, $\mathbf{W}_v \in \mathbb{R}^{D \times C}$. Here, s_j signifies a label distribution without the contribution of nt_j. Using this, the relevance score of each nt_j for the final prediction is given by the difference between the classifier score for the predicted label based on the entire input and the label score based on the input without nt_j: $\mathbf{r}_j = (l_Y)_i|_{i=P_C} - (s_j)_i|_{i=P_C}$, where $\mathbf{r}_j$ is the relevance score of the concept nt_j.

2.4 Global Interpretability layer (GIL)

The Global Interpretability Layer GIL aims to interpret each data sample $\mathbf{x}$ by providing a set of K concepts from the training data which most influenced the model's predictions. Such an approach is advantageous as we can now understand how important concepts from the training set influenced the model decision to predict the label of a new input, providing more granularity than methods that use entire samples from the training data for post-

[2] We experimented with different pooling strategies (mean pooling, sum pooling and pooled [CLS] token representation) and all of them performed similarly. We chose to use the pooled [CLS] token for the final model as this is the most commonly used method for representing the entire input.

hoc interpretability (Koh and Liang, 2017; Han et al., 2020).

We first build a *concept store* Q which holds all the concepts from the training data. Given model $\mathcal{M}$, we represent each concept candidate from the training data, q_k as a mean pooled representation of its constituent words $q_k = \frac{\sum_{w \in q_k} e(w)}{len(q_k)} \in \mathbb{R}^D$, where e represents the embedding layer of $\mathcal{M}$ and $len(q_k)$ represents the number of words in q_k. Q is represented by a set of $\{q\}_{1:N_Q}$, which are N_Q number of concepts from the training data. As the model $\mathcal{M}$ is finetuned for a downstream task, the representations q_k are constantly updated. Typically, we re-index all candidate representations q_k after every fixed number of training steps.

For any input $\mathbf{x}$, $\texttt{GIL}$ produces a set of K concepts $\{q\}_{1:K}$ from Q that are most influential as defined by the cosine similarity function:

$$d(\mathbf{x}, Q) = \frac{\mathbf{x} \cdot q}{\|\mathbf{x}\| \|q\|} \quad \forall q \in Q$$

Taking $\mathbf{u}_{\mathbb{S}}$ as input, $\texttt{GIL}$ uses dense inner product search to retrieve the top-K influential concepts $\mathcal{C}_G$ for the sample. Differentiable approaches through Maximum Inner Product Search (MIPS) has been shown to be effective in Question-Answering settings (Guu et al., 2020; Dhingra et al., 2020) to leverage retrieved knowledge for reasoning [3]. Motivated by this, we repurpose this retrieval approach to identify the influential concepts from the training data and learn it end-to-end via backpropagation. Our inner product model for $\texttt{GIL}$ is defined as follows:

$$p(q|\mathbf{x}_i) = \frac{exp\ d(\mathbf{u}_{\mathbb{S}}, q)}{\sum_{q'} exp\ d(\mathbf{u}_{\mathbb{S}}, q')}$$

2.5 Training

SELFEXPLAIN is trained to maximize the conditional log-likelihood of predicting the class at all the final layers: linear (for label prediction), $\texttt{LIL}$, and $\texttt{GIL}$. Regularizing models with explanation specific losses have been shown to improve inherently interpretable models (Melis and Jaakkola, 2018) for local interpretability. We extend this idea for both global and local interpretable output for our classifier model. For our training, we regularize the loss through $\texttt{GIL}$ and $\texttt{LIL}$ layers by optimizing their output for the end-task as well.

For the $\texttt{GIL}$ layer, we aggregate the scores over all the retrieved $q_{1:K}$ as a weighted sum, followed by an activation layer, linear layer and softmax to compute the log-likelihood loss as follows:

$$l_G = \texttt{softmax}(\mathbf{W}_u \times g(\sum_{k=1}^{K} \mathbf{w}_k \times q_k) + \mathbf{b}_u)$$

and $\mathcal{L}_G = -\sum_{c=1}^{C} y_c \log(l_G)$ where the global interpretable concepts are denoted by $\mathcal{C}_G = q_{1:K}$, $\mathbf{W}_u \in \mathbb{R}^{D \times C}$, $\mathbf{w}_k \in \mathbb{R}$ and g represents $relu$ activation, and l_G represents the softmax for the $\texttt{GIL}$ layer.

For the $\texttt{LIL}$ layer, we compute a weighted aggregated representation over s_j and compute the log-likelihood loss as follows:

$$l_L = \sum_{j, j \neq \mathbb{S}} \mathbf{w}_{sj} \times s_j, \ \mathbf{w}_{sj} \in \mathbb{R}$$

and $\mathcal{L}_L = -\sum_{c=1}^{C} y_c \log(l_L)$. To train the model, we optimize for the following joint loss,

$$\mathcal{L} = \alpha \times \mathcal{L}_G + \beta \times \mathcal{L}_L + \mathcal{L}_Y$$

where $\mathcal{L}_Y = -\sum_{c=1}^{C} y_c\, log(l_Y)$. Here, α and β are regularization hyper-parameters. All loss components use cross-entropy loss based on task label y_c.

3 Dataset and Experiments

Dataset	C	L	Train	Test
SST-2	2	19	68,222	1,821
SST-5	5	18	10,754	1,101
TREC-6	6	10	5,451	500
TREC-50	50	10	5,451	499
SUBJ	2	23	8,000	1,000

Table 1: Dataset statistics, where **C** is the number of classes and **L** is the average sentence length

Datasets: We evaluate our framework on five classification datasets: (i) SST-2 [4] Sentiment Classification task (Socher et al., 2013): the task is to predict the sentiment of movie review sentences as a binary classification task. (ii) SST-5 [5] : a fine-grained sentiment classification task that uses the

[3] MIPS can often be efficiently scaled using approximate algorithms (Shrivastava and Li, 2014)

[4] https://gluebenchmark.com/tasks

[5] https://nlp.stanford.edu/sentiment/index.html

Model	SST-2	SST-5	TREC-6	TREC-50	SUBJ
XLNet	93.4	53.8	**96.6**	82.8	96.2
SELFEXPLAIN-XLNet (K=5)	**94.6**	**55.2**	96.4	**83.0**	**96.4**
SELFEXPLAIN-XLNet (K=10)	94.4	55.2	96.4	82.8	96.4
RoBERTa	94.8	53.5	97.0	89.0	96.2
SELFEXPLAIN-RoBERTa (K=5)	**95.1**	**54.3**	**97.6**	**89.4**	**96.3**
SELFEXPLAIN-RoBERTa (K=10)	95.1	54.1	97.6	89.2	96.3

Table 2: Performance comparison of models with and without GIL and LIL layers. All experiments used the same encoder configurations. We use the development set for SST-2 results (test set of SST-2 is part of GLUE benchmark) and test sets for - SST-5, TREC-6, TREC-50 and SUBJ $\alpha, \beta = 0.1$ for all the above settings.

same dataset as before, but modifies it into a finer-grained 5-class classification task. (iii) TREC-6 [6] : a question classification task proposed by Li and Roth (2002), where each question should be classified into one of 6 question types. (iv) TREC-50: a fine-grained version of the same TREC-6 question classification task with 50 classes (v) SUBJ: subjective/objective binary classification dataset (Pang and Lee, 2005). The dataset statistics are shown in Table 1.

Experimental Settings: For our SELFEXPLAIN experiments, we consider two transformer encoder configurations as our base models: (1) RoBERTa encoder (Liu et al., 2019) — a robustly optimized version of BERT (Devlin et al., 2019). (2) XLNet encoder (Yang et al., 2019) — a transformer model based on Transformer-XL (Dai et al., 2019) architecture.

We incorporate SELFEXPLAIN into RoBERTa and XLNet, and use the above encoders without the GIL and LIL layers as the baselines. We generate parse trees (Kitaev and Klein, 2018) to extract target concepts for the input and follow same pre-processing steps as the original encoder configurations for the rest. We also maintain the hyperparameters and weights from the pre-training of the encoders. The architecture with GIL and LIL modules are fine-tuned on datasets described in §3. For the number of global influential concepts K, we consider two settings $K = 5, 10$. We also perform hyperparameter tuning on $\alpha, \beta = \{0.01, 0.1, 0.5, 1.0\}$ and report results on the best model configuration. All models were trained on an NVIDIA V-100 GPU.

Classification Results : We first evaluate the utility of classification models after incorporating GIL and LIL layers in Table 2. Across the different classification tasks, we observe that SELFEXPLAIN-RoBERTa and SELFEXPLAIN-XLNet consistently show competitive performance compared to the base models except for a marginal drop in TREC-6 dataset for SELFEXPLAIN-XLNet.

We also observe that the hyperparameter K did not make noticeable difference. Additional ablation experiments in Table 3 suggest that gains through GIL and LIL are complementary and both layers contribute to performance gains.

Model	Accuracy
XLNet-Base	93.4
SELFEXPLAIN-XLNet + LIL	94.3
SELFEXPLAIN-XLNet + GIL	94.0
SELFEXPLAIN-XLNet + GIL + LIL	94.6
RoBERTa-Base	94.8
SELFEXPLAIN-RoBERTa + LIL	94.8
SELFEXPLAIN-RoBERTa + GIL	94.8
SELFEXPLAIN-RoBERTa + GIL + LIL	95.1

Table 3: Ablation: SELFEXPLAIN-XLNet and SELFEXPLAIN-RoBERTa base models on SST-2.

4 Explanation Evaluation

Explanations are notoriously difficult to evaluate quantitatively (Doshi-Velez et al., 2017). A *good* model explanation should be (i) relevant to the current input and predictions and (ii) understandable to humans (DeYoung et al., 2020; Jacovi and Goldberg, 2020; Wiegreffe et al., 2020; Jain et al., 2020). Towards this, we evaluate whether the explanations along the following diverse criteria:

- **Sufficiency** – Do explanations sufficiently reflect the model predictions?
- **Plausibility** – Do explanations appear plausible and understandable to humans?

[6] https://cogcomp.seas.upenn.edu/Data/QA/QC/

- **Trustability** – Do explanations improve human trust in model predictions?

From SELFEXPLAIN, we extracted (i) *Most relevant local concepts*: these are the top ranked phrases based on $\mathbf{r}(nt)_{1:J}$ from the `LIL` layer and (ii) *Top influential global concepts:* these are the most influential concepts $q_{1:K}$ ranked by the output of `GIL` layer as the model explanations to be used for evaluations.

4.1 Do SELFEXPLAIN explanations reflect predicted labels?

Sufficiency aims to evaluate whether model explanations alone are highly indicative of the predicted label (Jacovi et al., 2018; Yu et al., 2019). "Faithfulness-by-construction" (FRESH) pipeline (Jain et al., 2020) is an example of such framework to evaluate sufficiency of explanations: the sole explanations, without the remaining parts of the input, must be sufficient for predicting a label. In FRESH, a BERT (Devlin et al., 2019) based classifier is trained to perform a task using only the extracted explanations without the rest of the input. An explanation that achieves high accuracy using this classifier is indicative of its ability to recover the original model prediction.

We evaluate the explanations on the sentiment analysis task. Explanations from SELFEXPLAIN are incorporated to the FRESH framework and we compare the predictive accuracy of the explanations in comparison to baseline explanation methods. Following Jain et al. (2020), we use the same experimental setup and saliency-based baselines such as attention (Lei et al., 2016; Bastings et al., 2019) and gradient (Li et al., 2016) based explanation methods. From Table 4[7], we observe that SELFEXPLAIN explanations from `LIL` and `GIL` show high predictive performance compared to all the baseline methods. Additionally, `GIL` explanations outperform full-text (an explanation that uses all of the input sample) performance, which is often considered an upper-bound for span-based explanation approaches. We hypothesize that this is because `GIL` explanation concepts from the training data are very relevant to help disambiguate the input text. In summary, outputs from SELFEXPLAIN are more predictive of the label compared to prior explanation methods indicating higher sufficiency of explanations.

Model	Explanation	Accuracy
Full input text	-	0.90
Lei et al. (2016)	contiguous	0.71
	top-K tokens	0.74
Bastings et al. (2019)	contiguous	0.60
	top-K tokens	0.59
Li et al. (2016)	contiguous	0.70
	top-K tokens	0.68
`[CLS]` Attn	contiguous	0.81
	top-K tokens	0.81
SELFEXPLAIN-`LIL`	top-K concepts	**0.84**
SELFEXPLAIN-`GIL`	top-K concepts	**0.93**

Table 4: Model predictive performances (prediction accuracy) on SST-dataset test set. *Contiguous* refers to explanations that are spans of text and top-K refers to model-ranked top-K tokens. SELFEXPLAIN also uses at most top-K (where K=2) concepts for both `LIL` and `GIL`. SELFEXPLAIN explanations from both `GIL` and `LIL` outperform all baselines.

4.2 Are SELFEXPLAIN explanations plausible and trustable for humans?

Human evaluation is commonly used to evaluate *plausibility* and *trustability*. To this end, 14 human judges[8] annotated 50 samples from the SST-2 validation set of sentiment excerpts (Socher et al., 2013). Each judge compared local and global explanations produced by the SELFEXPLAIN-XLNet model against two commonly used interpretability methods (i) Influence functions (Han et al., 2020) for global interpretability and (ii) Saliency detection (Simonyan et al., 2014) for local interpretability. We follow a setup discussed in Han et al. (2020). Each judge was provided the evaluation criteria (detailed next) with a corresponding description. The models to be evaluated were anonymized and humans were asked to rate them according to the evaluation criteria alone.

Following Ehsan et al. (2019), we analyse the *plausibility* of explanations which aims to understand how users would perceive such explanations if they were generated by humans. We adopt two criteria proposed by Ehsan et al. (2019):

Adequate justification : Adequately justifying the prediction is considered to be an important criteria for acceptance of a model (Davis, 1989). We evaluate the *adequacy* of the explanation by

[7]In these experiments, explanations are pruned at a maximum of 20% of input. For SELFEXPLAIN, we select upto top-K concepts thresholding at 20% of input

[8]Annotators are graduate students in computer science.

Sample	P_C	Top relevant phrases from LIL	Top influential concepts from `GIL`
the iditarod lasts for days - this just felt like it did .	neg	for days	exploitation piece, heart attack
corny, schmaltzy and predictable, but still manages to be kind of heart warming, nonetheless.	pos	corny, schmaltzy, of heart	successfully blended satire, spell binding fun
suffers from the lack of a compelling or comprehensible narrative .	neg	comprehensible, the lack of	empty theatres, tumble weed
the structure the film takes may find matt damon and ben affleck once again looking for residuals as this officially completes a good will hunting trilogy that was never planned .	pos	the structure of the film	bravo, meaning and consolation

Table 5: Sample output from the model and its corresponding local and global interpretable outputs SST-2 (P_C stands for predicted class) (some input text cut for brevity). More qualitative examples in appendix §A.2

asking human judges: "Does the explanation adequately justifies the model prediction?" Participants deemed explanations that were irrelevant or incomplete as less adequately justifying the model prediction. Human judges were shown the following: (i) input, (ii) gold label, (iii) predicted label, and (iv) explanations from baselines and SELFEXPLAIN. The models were anonymized and shuffled.

Figure 3 (left) shows that SELFEXPLAIN achieves a gain of 32% in perceived adequate justification, providing further evidence that humans perceived SELFEXPLAIN explanations as more plausible compared to the baselines.

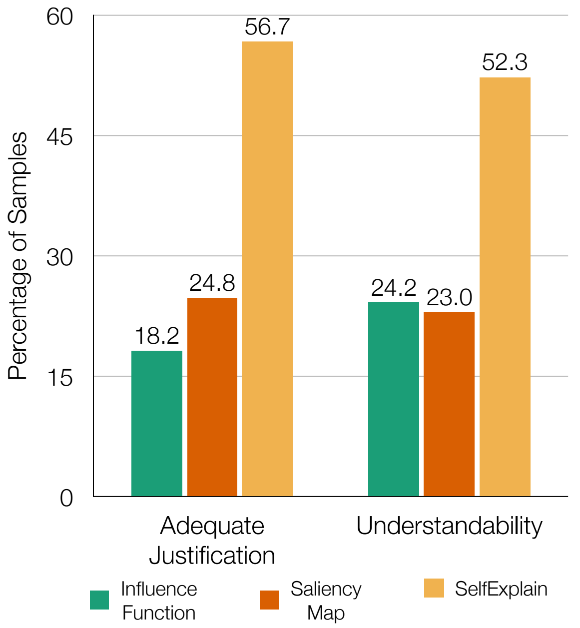

Figure 3: *Adequate justification* and *understandability* of SELFEXPLAIN against baselines. The vertical axis shows the percentage of samples evaluated by humans. Humans judge SELFEXPLAIN explanations to better justify the predictions and be more understandable.

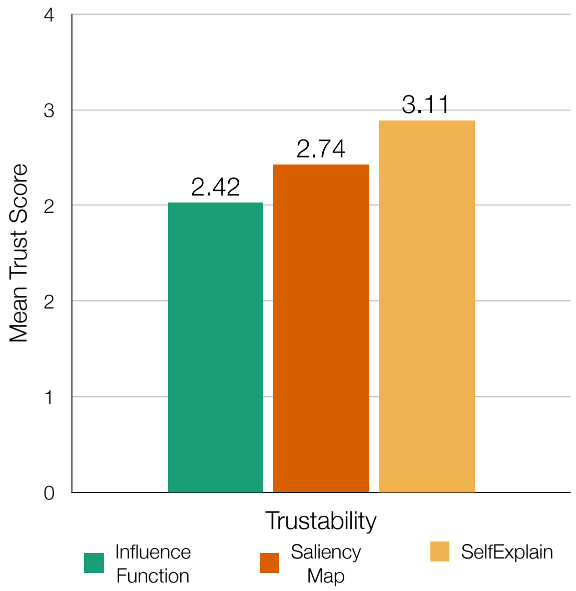

Figure 4: *Mean trust score* of SELFEXPLAIN against baselines. The vertical axis show mean trust labeled on 1-5 likert scale. Humans judge SELFEXPLAIN explanations improve trust in model predictions.

Understandability: An essential criterion for transparency in an AI system is the ability of a user to *understand* model explanations (Doshi-Velez et al., 2017). Our understandability metric evaluates whether a human judge can understand the explanations presented by the model, which would equip a non-expert to verify the model predictions. Human judges were presented (i) the input, (ii) gold label, (iii) sentiment label prediction, and (iv) explanations from different methods (baselines, and SELFEXPLAIN), and were asked to select the explanation that they perceived to be more understandable. Figure 3 (right) shows that SELFEXPLAIN achieves 29% improvement over the best-performing baseline in terms of understandability of the model explanation.

Trustability: In addition to plausibility, we also evaluate user *trust* of the explanations (Singh et al., 2019; Jin et al., 2020). To evaluate user trust, We follow the same experimental setup as Singh et al. (2019) and Jin et al. (2020) to compute the *mean trust score*. For each data sample, subjects were shown explanations and the model prediction from the three interpretability methods and were asked to rate on a Likert scale of 1–5 based on how much trust did each of the model explanations instill. Figure 4 shows the mean-trust score of SELFEXPLAIN in comparison to the baselines. We observe from the results that concept-based explanations are perceived more trustworthy for humans.

5 Analysis

Table 5 shows example interpretations by SELFEXPLAIN; we show some additional analysis of explanations from SELFEXPLAIN[9] in this section.

Does SELFEXPLAIN's explanation help predict model behavior? In this setup, humans are presented with an explanation and an input, and must correctly predict the model's output (Doshi-Velez and Kim, 2017; Lertvittayakumjorn and Toni, 2019; Hase and Bansal, 2020). We randomly selected 16 samples spanning equal number of true positives, true negatives, false positives and false negatives from the dev set. Three human judges were tasked to predict the model decision with and without the presence of model explanation. We observe that when users were presented with the explanation, their ability to predict model decision improved by an average of 22%, showing that with SELFEXPLAIN's explanations, humans could better understand model's behavior.

Performance Analysis: In GIL, we study the performance trade-off of varying the number of retrieved influential concepts K. From a performance perspective, there is only marginal drop in moving from the base model to SELFEXPLAIN model with both GIL and LIL (shown in Table 6). From our experiments with human judges, we found that for sentence level classification tasks $K = 5$ is preferable for a balance of performance and the ease of interpretability.

GIL top-K	steps/sec	memory
base	2.74	$1x$
K=5*	2.50	$1.03x$
K=100	2.48	$1.04x$
K=1000	2.20	$1.07x$

Table 6: Effect of K from GIL. We use SELFEXPLAIN-XLNet on SST-2 for this analysis. *K=1/5/10 did not show considerable difference among them

LIL-GIL-Linear layer agreement: To understand whether our explanations lead to predicting the same label as the model's prediction, we analyze whether the final logits activations on the GIL and LIL layers agree with the linear layer activations. Towards this, we compute an agreement between label distributions from GIL and LIL layers to the distribution of the linear layer. Our LIL-*linear* F1 is 96.6%, GIL-*linear* F1 100% and GIL-LIL-*linear* F1 agreement is 96.6% for SELFEXPLAIN-XLNet on the SST-2 dataset. We observe that the agreement rates between the GIL , LIL and the linear layer are very high, validating that SELFEXPLAIN's layers agree on the same model classification prediction, showing that GIL and LIL concepts lead to same predictions.

Are LIL concepts relevant? For this analysis, we randomly selected 50 samples from SST2 dev set and removed the top most salient phrases ranked by LIL. Annotators were asked to predict the label without the most relevant local concept and the accuracy dropped by 7%. We also computed the SELFEXPLAIN-XLNet classifier's accuracy on the same input and the accuracy dropped by $\sim$14%.[10] This suggests that LIL captures relevant local concepts.[11]

Stability: do similar examples have similar explanations? Melis and Jaakkola (2018) argue that a crucial property that interpretable models need to address is *stability*, where the model should be robust enough that a minimal change in the input should not lead to drastic changes in the observed interpretations. We qualitatively analyze this by measuring the overlap of SELFEXPLAIN's extracted concepts for similar examples. Table 8 shows a representative example in which minor variations in the input lead to differently ranked

[9]additional analysis in appendix due to space constraints

[10]Statistically significant by Wilson interval test.

[11]Samples from this experiment are shown in §A.3.

Input	Top LIL interpretations	Top GIL interpretations
it 's a very charming and often affecting journey	often affecting, very charming	scenes of cinematic perfection that steal your heart away, submerged, that extravagantly
it ' s a charming and often affecting journey of people	of people, charming and often affecting	scenes of cinematic perfection that steal your heart away, submerged, that extravagantly

Table 7: Sample (from SST-2) of an input perturbation lead to different local concepts, but global concepts remain stable.

local phrases, but their global influential concepts remain stable.

6 Related Work

Post-hoc Interpretation Methods: Predominant based methods for post-hoc interpretability in NLP use gradient based methods (Simonyan et al., 2014; Sundararajan et al., 2017; Smilkov et al., 2017). Other post-hoc interpretability methods such as Singh et al. (2019) and Jin et al. (2020) decompose relevant and irrelevant aspects from hidden states and obtain a relevance score. While the methods above focus on local interpretability, works such as Han et al. (2020) aim to retrieve influential training samples for global interpretations. Global interpretability methods are useful not only to facilitate explainability, but also to detect and mitigate artifacts in data (Pezeshkpour et al., 2021; Han and Tsvetkov, 2021).

Inherently Intepretable Models: Heat maps based on attention (Bahdanau et al., 2014) are one of the commonly used interpretability tools for many downstream tasks such as machine translation (Luong et al., 2015), summarization (Rush et al., 2015) and reading comprehension Hermann et al. (2015). Another recent line of work explores collecting *rationales* (Lei et al., 2016) through expert annotations (Zaidan and Eisner, 2008). Notable work in collecting external rationales include Cos-E (Rajani et al., 2019), e-SNLI (Camburu et al., 2018) and recently, Eraser benchmark (DeYoung et al., 2020). Alternative lines of work in this class of models include Card et al. (2019) that relies on interpreting a given sample as a weighted sum of the training samples while Croce et al. (2019) identifies influential training samples using a kernel-based transformation function. Jiang and Bansal (2019) produce interpretations of a given sample through modular architectures, where model decisions are explained through outputs of intermediate modules. A class of inherently interpretable classifiers explain model predictions locally using human-understandable high-level *concepts* such as prototypes (Melis and Jaakkola, 2018; Chen et al., 2019) and interpretable classes (Koh et al., 2020; kuan Yeh et al., 2020). They were recently proposed for computer vision applications, but despite their promise have not yet been adopted in NLP.

7 Conclusion

In this paper, we propose SELFEXPLAIN, a novel self-explaining framework that enables explanations through higher-level concepts, improving from low-level word attributions. SELFEXPLAIN provides both local explanations (via relevance of each input concept) and global explanations (through influential concepts from the training data) in a single framework via two novel modules (LIL and GIL), and trainable end-to-end. Through human evaluation, we show that our interpreted model outputs are perceived as more trustworthy, understandable, and adequate for explaining model decisions compared to previous approaches to explainability. This opens an exciting research direction for building inherently interpretable models for text classification. Future work will extend the framework to other tasks and to longer contexts, beyond single input sentence.

Acknowledgements

This material is based upon work funded by the DARPA CMO under Contract No. HR001120C0124, and by the United States Department of Energy (DOE) National Nuclear Security Administration (NNSA) Office of Defense Nuclear Nonproliferation Research and Development (DNN R&D) Next-Generation AI research portfolio. The views and opinions of authors expressed herein do not necessarily state or reflect those of the United States Government or any agency thereof.

References

Sercan Ö. Arik and T. Pfister. 2020. Protoattend: Attention-based prototypical learning. *J. Mach. Learn. Res.*, 21:210:1–210:35.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. In *ICLR*.

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.

Oana-Maria Camburu, Tim Rocktäschel, Thomas Lukasiewicz, and Phil Blunsom. 2018. e-snli: Natural language inference with natural language explanations. In *NeurIPS*.

Dallas Card, Michael Zhang, and Noah A Smith. 2019. Deep weighted averaging classifiers. In *Proceedings of the Conference on Fairness, Accountability, and Transparency*, pages 369–378.

Chaofan Chen, Oscar Li, Daniel Tao, Alina Barnett, Cynthia Rudin, and Jonathan K Su. 2019. This looks like that: deep learning for interpretable image recognition. In *Advances in neural information processing systems*, pages 8930–8941.

Danilo Croce, Daniele Rossini, and Roberto Basili. 2019. Auditing deep learning processes through kernel-based explanatory models. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 4028–4037.

Zihang Dai, Zhilin Yang, Yiming Yang, Jaime Carbonell, Quoc V Le, and Ruslan Salakhutdinov. 2019. Transformer-xl: Attentive language models beyond a fixed-length context. *arXiv preprint arXiv:1901.02860*.

Fred D. Davis. 1989. Perceived usefulness, perceived ease of use, and user acceptance of information technology. *MIS Q.*, 13:319–340.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *NAACL 2019*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Jay DeYoung, Sarthak Jain, Nazneen Fatema Rajani, Eric Lehman, Caiming Xiong, Richard Socher, and Byron C. Wallace. 2020. ERASER: A benchmark to evaluate rationalized NLP models. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4443–4458, Online. Association for Computational Linguistics.

Bhuwan Dhingra, Manzil Zaheer, Vidhisha Balachandran, Graham Neubig, Ruslan Salakhutdinov, and William W. Cohen. 2020. Differentiable reasoning over a virtual knowledge base. In *International Conference on Learning Representations*.

Finale Doshi-Velez and Been Kim. 2017. Towards a rigorous science of interpretable machine learning. *arXiv preprint arXiv:1702.08608*.

Finale Doshi-Velez, Mason Kortz, Ryan Budish, Chris Bavitz, Sam Gershman, D. O'Brien, Stuart Schieber, J. Waldo, D. Weinberger, and Alexandra Wood. 2017. Accountability of ai under the law: The role of explanation. *ArXiv*, abs/1711.01134.

Upol Ehsan, Pradyumna Tambwekar, Larry Chan, Brent Harrison, and Mark O Riedl. 2019. Automated rationale generation: a technique for explainable ai and its effects on human perceptions. In *Proceedings of the 24th International Conference on Intelligent User Interfaces*, pages 263–274.

Suchin Gururangan, Swabha Swayamdipta, Omer Levy, Roy Schwartz, Samuel Bowman, and Noah A. Smith. 2018. Annotation artifacts in natural language inference data. In *NAACL 201*, pages 107–112, New Orleans, Louisiana. Association for Computational Linguistics.

Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, and Ming-Wei Chang. 2020. Realm: Retrieval-augmented language model pre-training. *arXiv preprint arXiv:2002.08909*.

Xiaochuang Han and Yulia Tsvetkov. 2021. Influence tuning: Demoting spurious correlations via instance attribution and instance-driven updates. In *Findings of EMNLP*.

Xiaochuang Han, Byron C. Wallace, and Yulia Tsvetkov. 2020. Explaining black box predictions and unveiling data artifacts through influence functions. In *ACL*.

Peter Hase and Mohit Bansal. 2020. Evaluating explainable AI: Which algorithmic explanations help users predict model behavior? In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5540–5552, Online. Association for Computational Linguistics.

Karl Moritz Hermann, Tomas Kocisky, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. 2015. Teaching machines to read and comprehend. In C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, editors, *Advances in Neural Information Processing Systems 28*, pages 1693–1701. Curran Associates, Inc.

Alon Jacovi and Yoav Goldberg. 2020. Towards faithfully interpretable NLP systems: How should we define and evaluate faithfulness? In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4198–4205, Online. Association for Computational Linguistics.

Alon Jacovi, Oren Sar Shalom, and Y. Goldberg. 2018. Understanding convolutional neural networks for text classification. *ArXiv*, abs/1809.08037.

Sarthak Jain and Byron C. Wallace. 2019. Attention is not Explanation. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 3543–3556, Minneapolis, Minnesota. Association for Computational Linguistics.

Sarthak Jain, Sarah Wiegreffe, Yuval Pinter, and Byron C. Wallace. 2020. Learning to faithfully rationalize by construction. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4459–4473, Online. Association for Computational Linguistics.

Yichen Jiang and Mohit Bansal. 2019. Self-assembling modular networks for interpretable multi-hop reasoning. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 4474–4484, Hong Kong, China. Association for Computational Linguistics.

Xisen Jin, Zhongyu Wei, Junyi Du, Xiangyang Xue, and Xiang Ren. 2020. Towards hierarchical importance attribution: Explaining compositional semantics for neural sequence models. In *International Conference on Learning Representations*.

Been Kim, Cynthia Rudin, and Julie A Shah. 2014. The bayesian case model: A generative approach for case-based reasoning and prototype classification. In *Advances in neural information processing systems*, pages 1952–1960.

Nikita Kitaev and D. Klein. 2018. Constituency parsing with a self-attentive encoder. In *ACL*.

Pang Wei Koh and Percy Liang. 2017. Understanding black-box predictions via influence functions. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 1885–1894. JMLR. org.

Pang Wei Koh, Thao Nguyen, Yew Siang Tang, Stephen Mussmann, Emma Pierson, Been Kim, and Percy Liang. 2020. Concept bottleneck models. *NeurIPS*.

Chih kuan Yeh, Been Kim, Sercan Arik, Chun-Liang Li, Pradeep Ravikumar, and Tomas Pfister. 2020. On completeness-aware concept-based explanations in deep neural networks.

Sachin Kumar, Shuly Wintner, Noah A. Smith, and Yulia Tsvetkov. 2019. Topics to avoid: Demoting latent confounds in text classification. In *Proc. EMNLP*, pages 4151–4161.

Tao Lei, Regina Barzilay, and Tommi Jaakkola. 2016. Rationalizing neural predictions. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 107–117, Austin, Texas. Association for Computational Linguistics.

Piyawat Lertvittayakumjorn and Francesca Toni. 2019. Human-grounded evaluations of explanation methods for text classification. In *EMNLP/IJCNLP*.

J. Li, Xinlei Chen, E. Hovy, and Dan Jurafsky. 2016. Visualizing and understanding neural models in nlp. In *HLT-NAACL*.

Xin Li and Dan Roth. 2002. Learning question classifiers. In *Proceedings of the 19th international conference on Computational linguistics-Volume 1*, pages 1–7. Association for Computational Linguistics.

Zachary C Lipton. 2018. The mythos of model interpretability. *Queue*, 16(3):31–57.

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

Thang Luong, Hieu Pham, and Christopher D. Manning. 2015. Effective approaches to attention-based neural machine translation. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1412–1421, Lisbon, Portugal. Association for Computational Linguistics.

R. Thomas McCoy, Ellie Pavlick, and Tal Linzen. 2019. Right for the wrong reasons: Diagnosing syntactic heuristics in natural language inference. In *Proc. ACL*.

David Alvarez Melis and Tommi Jaakkola. 2018. Towards robust interpretability with self-explaining neural networks. In *Advances in Neural Information Processing Systems*, pages 7775–7784.

Richard Montague. 1970. English as a formal language. In Bruno Visentini, editor, *Linguaggi nella societa e nella tecnica*, pages 188–221. Edizioni di Communita.

Grégoire Montavon, Sebastian Lapuschkin, Alexander Binder, W. Samek, and K. Müller. 2017. Explaining nonlinear classification decisions with deep taylor decomposition. *Pattern Recognit.*, 65:211–222.

Bo Pang and Lillian Lee. 2005. Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales. *arXiv preprint cs/0506075*.

Pouya Pezeshkpour, Sarthak Jain, Sameer Singh, and Byron C Wallace. 2021. Combining feature and instance attribution to detect artifacts. *arXiv preprint arXiv:2107.00323*.

Danish Pruthi, Mansi Gupta, Bhuwan Dhingra, Graham Neubig, and Zachary C Lipton. 2020. Learning to deceive with attention-based explanations. In *ACL*.

Nazneen Fatema Rajani, Bryan McCann, Caiming Xiong, and Richard Socher. 2019. Explain yourself! leveraging language models for commonsense reasoning. *ACL*.

Marco Ribeiro, Sameer Singh, and Carlos Guestrin. 2016. "why should I trust you?": Explaining the predictions of any classifier. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Demonstrations*, pages 97–101, San Diego, California. Association for Computational Linguistics.

Cynthia Rudin. 2019. Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nature Machine Intelligence*, 1(5):206–215.

Alexander M. Rush, Sumit Chopra, and Jason Weston. 2015. A neural attention model for abstractive sentence summarization. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 379–389, Lisbon, Portugal. Association for Computational Linguistics.

Sofia Serrano and Noah A. Smith. 2019. Is attention interpretable? In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2931–2951, Florence, Italy. Association for Computational Linguistics.

Avanti Shrikumar, Peyton Greenside, and Anshul Kundaje. 2017. Learning important features through propagating activation differences. volume 70 of *Proceedings of Machine Learning Research*, pages 3145–3153, International Convention Centre, Sydney, Australia. PMLR.

Anshumali Shrivastava and Ping Li. 2014. Asymmetric lsh (alsh) for sublinear time maximum inner product search (mips). In *Advances in Neural Information Processing Systems*, pages 2321–2329.

Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. 2014. Deep inside convolutional networks: Visualising image classification models and saliency maps. *CoRR*, abs/1312.6034.

Chandan Singh, W. James Murdoch, and Bin Yu. 2019. Hierarchical interpretations for neural network predictions. In *International Conference on Learning Representations*.

Daniel Smilkov, Nikhil Thorat, Been Kim, Fernanda Viégas, and Martin Wattenberg. 2017. Smoothgrad: removing noise by adding noise. *arXiv preprint arXiv:1706.03825*.

Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D Manning, Andrew Y Ng, and Christopher Potts. 2013. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 conference on empirical methods in natural language processing*, pages 1631–1642.

Tony Sun, Andrew Gaut, Shirlyn Tang, Yuxin Huang, Mai ElSherief, Jieyu Zhao, Diba Mirza, Elizabeth Belding, Kai-Wei Chang, and William Yang Wang. 2019. Mitigating gender bias in natural language processing: Literature review. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1630–1640, Florence, Italy. Association for Computational Linguistics.

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.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.

Sarah Wiegreffe, Ana Marasović, and Noah A. Smith. 2020. Measuring association between labels and free-text rationales. *ArXiv*, abs/2010.12762.

Sarah Wiegreffe and Yuval Pinter. 2019. Attention is not not explanation. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 11–20, Hong Kong, China. Association for Computational Linguistics.

Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron C. Courville, R. Salakhutdinov, R. Zemel, and Yoshua Bengio. 2015. Show, attend and tell: Neural image caption generation with visual attention. In *ICML*.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R Salakhutdinov, and Quoc V Le. 2019. Xlnet: Generalized autoregressive pretraining for language understanding. In *Advances in neural information processing systems*, pages 5754–5764.

Mo Yu, S. Chang, Y. Zhang, and T. Jaakkola. 2019. Rethinking cooperative rationalization: Introspective extraction and complement control. *ArXiv*, abs/1910.13294.

Omar Zaidan and Jason Eisner. 2008. Modeling annotators: A generative approach to learning from annotator rationales. In *Proceedings of the 2008 conference on Empirical methods in natural language processing*, pages 31–40.

Jieyu Zhao, Tianlu Wang, Mark Yatskar, Vicente Ordonez, and Kai-Wei Chang. 2017. Men also like shopping: Reducing gender bias amplification using corpus-level constraints. In *Proc. of EMNLP*, pages 2979–2989.

A Appendix

A.1 Additional Analysis

Stability: do similar examples have similar explanations? Melis and Jaakkola (2018) argue that a crucial property that interpretable models need to address is *stability*, where the model should be robust enough that a minimal change in the input should not lead to drastic changes in the observed interpretations. We qualitatively analyze this by measuring the overlap of SELFEXPLAIN's extracted concepts for similar examples. Table 8 shows a representative example in which minor variations in the input lead to differently ranked local phrases, but their global influential concepts remain stable.

A.2 Qualitative Examples

Table 9 shows some qualitative examples from our best performing SST-2 model.

A.3 Relevant Concept Removal

Table 10 shows us the samples where the model flipped the label after the most relevant local concept was removed. In this table, we show the original input, the perturbed input after removing the most relevant local concept, and the corresponding model predictions.

Input	Top LIL interpretations	Top GIL interpretations
it 's a very charming and often affecting journey	often affecting, very charming	scenes of cinematic perfection that steal your heart away, submerged, that extravagantly
it ' s a charming and often affecting journey of people	of people, charming and often affecting	scenes of cinematic perfection that steal your heart away, submerged, that extravagantly

Table 8: Sample (from SST-2) of an input perturbation lead to different local concepts, but global concepts remain stable.

Input Sentence	Explanation from Input	Explanation from Training Data
offers much to enjoy ... and a lot to mull over in terms of love , loyalty and the nature of staying friends .	['much to enjoy', 'to enjoy', 'to mull over']	['feel like you ate a reeses without the peanut butter']
puts a human face on a land most westerners are unfamiliar with .	['put s a human face on a land most westerners are unfamiliar with', 'a human face']	['dazzle and delight us']
nervous breakdowns are not entertaining .	['n erv ous breakdown s', 'are not entertaining']	['mesmerizing portrait']
too slow , too long and too little happens .	['too long', 'too little happens', 'too little']	['his reserved but existential poignancy', 'very moving and revelatory footnote']
very bad .	['very bad']	['held my interest precisely', 'intriguing , observant', 'held my interest']
it haunts , horrifies , startles and fascinates ; it is impossible to look away .	['to look away', 'look away', 'it haun ts , horr ifies , start les and fasc inates']	['feel like you ate a reeses without the peanut butter']
it treats women like idiots .	['treats women like idiots', 'like idiots']	['neither amusing nor dramatic enough to sustain interest']
the director knows how to apply textural gloss , but his portrait of sex-as-war is strictly sitcom .	['the director', 'his portrait of sex - as - war']	['absurd plot twists' , 'idiotic court maneuvers and stupid characters']
too much of the humor falls flat .	['too much of the humor', 'too much', 'falls flat']	['infuriating']
the jabs it employs are short , carefully placed and dead-center .	['it employs', 'carefully placed', 'the j abs it employs']	['with terrific flair']
the words , ' frankly , my dear , i do n't give a damn , have never been more appropriate .	["do n 't give a damn"]	['spiteful idiots']
one of the best films of the year with its exploration of the obstacles to happiness faced by five contemporary individuals ... a psychological masterpiece .	['of the best films of the year', 'of the year', 'the year']	['bang']
my wife is an actress is an utterly charming french comedy that feels so american in sensibility and style it 's virtually its own hollywood remake .	['an utterly charming french comedy', 'utterly charming', 'my wife']	['all surface psychodramatics']

Table 9: Samples from SELFEXPLAIN's interpreted output.

Original Input	Perturbed Input	Original Prediction	Perturbed Prediction
unflinchingly bleak and desperate	unflinch ________________	negative	positive
the acting , costumes , music , cinematography and sound are all astounding given the production 's austere locales .	________ , costumes , music , cinematography and sound are all astounding given the production 's austere locales .	positive	negative
we root for (clara and paul) , even like them , though perhaps it 's an emotion closer to pity .	we root for (clara and paul) ,___________ , though perhaps it 's an emotion closer to pity .	positive	negative
the emotions are raw and will strike a nerve with anyone who 's ever had family trauma .	__________ are raw and will strike a nerve with anyone who 's ever had family trauma .	positive	negative
holden caulfield did it better .	holden caulfield __________ .	negative	positive
it 's an offbeat treat that pokes fun at the democratic exercise while also examining its significance for those who take part .	it 's an offbeat treat that pokes fun at the democratic exercise while also examining _________ for those who take part .	positive	negative
as surreal as a dream and as detailed as a photograph , as visually dexterous as it is at times imaginatively overwhelming .	_______________ and as detailed as a photograph , as visually dexterous as it is at times imaginatively overwhelming .	positive	negative
holm ... embodies the character with an effortlessly regal charisma .	holm ... embodies the character with ____________	positive	negative
it 's hampered by a lifetime-channel kind of plot and a lead actress who is out of her depth .	it 's hampered by a lifetime-channel kind of plot and a lead actress who is ____________ .	negative	negative

Table 10: Samples where the model predictions flipped after removing the most relevant local concept.