Towards Prediction Explainability through Sparse Communication

Marcos V. Treviso
Instituto de Telecomunicações
Lisboa, Portugal
marcosvtreviso@gmail.com

André F. T. Martins
Instituto de Telecomunicações & Unbabel
Lisboa, Portugal
andre.t.martins@tecnico.ulisboa.pt

Abstract

Explainability is a topic of growing importance in NLP. In this work, we provide a unified perspective of explainability as a communication problem between an explainer and a layperson about a classifier's decision. We use this framework to compare several prior approaches for extracting explanations, including gradient methods, representation erasure, and attention mechanisms, in terms of their communication success. In addition, we reinterpret these methods at the light of classical feature selection, and we use this as inspiration to propose new embedded methods for explainability, through the use of selective, sparse attention. Experiments in text classification, natural language entailment, and machine translation, using different configurations of explainers and laypeople (including both machines and humans), reveal an advantage of attention-based explainers over gradient and erasure methods. Furthermore, human evaluation experiments show promising results with post-hoc explainers trained to optimize communication success and faithfulness.

1 Introduction

The widespread use of machine learning systems to assist humans in decision making brings the need for providing interpretations for models' predictions (Lipton, 2018; Doshi-Velez and Kim, 2017; Rudin, 2019; Miller, 2019). This poses a challenge in NLP, where current state-of-the-art systems are based on deep neural networks and generally lack transparency (Goldberg, 2017; Peters et al., 2018; Devlin et al., 2019). Despite the large body of recent work in explainability (see §7 for a review), a unified perspective taking into account the human-machine interaction—a *communication* process in its essence—is still missing.

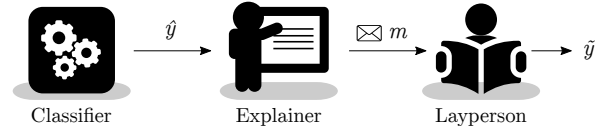Many methods have been proposed to generate explanations. Some neural network architectures

Figure 1: Our framework to model explainability as communication. Predictions $\hat{y}$ are made by a classifier C; an explainer E (either embedded in C or operating post-hoc) accesses these predictions and communicates an explanation (a message m) to the layperson L. Success of the communication is dictated by the ability of L and C to match their predictions: $\tilde{y} \stackrel{?}{=} \hat{y}$. Both the explainer and layperson can be humans or machines.

are equipped with built-in components—attention mechanisms—which weigh the relevance of input features for triggering a decision (Bahdanau et al., 2015; Vaswani et al., 2017). As a by-product, these weights provide plausible, but not always faithful, explanations (Jain and Wallace, 2019; Serrano and Smith, 2019; Wiegreffe and Pinter, 2019). Other approaches seek local explanations by evaluating the gradient of the predicted label with respect to the input features (Li et al., 2016a; Arras et al., 2017), or in a post-hoc manner by training a sparse linear model on a vicinity of the input example (Ribeiro et al., 2016) or by repeatedly querying the classifier with leave-one-out strategies (Li et al., 2016a; Feng et al., 2018).

How to assess the effectiveness of these different approaches? Several diagnostic tests have been proposed in prior work: Jain and Wallace (2019) assessed the explanatory power of attention weights by measuring their correlation with input gradients; Wiegreffe and Pinter (2019) and DeYoung et al. (2019) developed more informative tests, including a combination of comprehensiveness and sufficiency metrics and the correlation with human rationales. While useful, these frameworks rely on correlations and counterfactual simulation, and they do not provide a unified view of the existing

methods. They sidestep the main practical goal of prediction explainability—the ability to *communicate* an explanation to a human user.

In this work, we fill the gap above by proposing a unified framework that regards explainability as a **communication problem**: how can a machine communicate a justification for its decision (either a faithful explanation or a post-hoc interpretation) to a human? Our framework is inspired by human-grounded evaluation through **forward simulation/prediction**, as proposed by Doshi-Velez and Kim (2017, §3.2), where humans are presented with an explanation and an input, and must correctly simulate the model's output (regardless of the true output). We simulate this process by considering the interaction between a *classifier* (the original model whose predictions we want to explain), an *explainer* (which provides the explanations), and a *layperson* (which must recover the classifier's prediction). We experiment with explainers and laypeople being both humans and machines. Our framework allows comparing several prior approaches as different explainers, and inspires new ones: we propose explainers based on **selective attention** using sparsity (Martins and Astudillo, 2016; Peters et al., 2019), and a technique inspired by emergent communication (Foerster et al., 2016; Havrylov and Titov, 2017) that trains the explainer and layperson jointly.

Overall, our contributions are as follows:

- We draw a link between recent techniques for explainability of neural networks and classic feature selection in linear models (§2), leading to new embedded methods for explainability through selective, sparse attention (§3).

- We propose a new framework to assess explanatory power as the communication success rate between an explainer and a layperson (§4).

- We experiment with text classification, entailment, and machine translation, using different configurations of explainers and laypeople (humans or machines). This allows comparing in the same grounds models that use attention-based interpretations, post-hoc explanations, and gradient-based information (§5).

2 Static and Dynamic Feature Selection

A common way of generating explanations is by highlighting a small set of features (*rationales,*

Zaidan and Eisner (2008)) that are considered relevant to the model's decision (Lei et al., 2016; Ribeiro et al., 2016). A motivation for keeping this set small is the principle of parsimony ("Occam's razor"), according to which simple explanations should be preferred over complex ones. The same principle has inspired a large body of work in sparse modeling and feature selection, although the connection between these two fields of research has been somewhat overlooked.

Traditional feature selection methods (Guyon and Elisseeff, 2003) are mostly concerned with **model interpretability**, where the goal is to understand how the model behaves globally. In this framework, feature selection happens *statically* during model training, and features that are deemed irrelevant are permanently deleted from the model. By contrast, most recent work in explaining neural networks address **prediction explainability**, where feature selection happens *dynamically* at run time. Since explanations are input-dependent, features are never removed from the model (a feature that is not relevant for a particular input can be relevant for another).

Are these two worlds far away? Interestingly, the following typology, which originates from static feature selection (Guyon and Elisseeff, 2003, §4), still matches many existing approaches to *dynamic* feature selection (concrete examples in Table 1). The different methods are distinguished by the way they model the interaction between their main two components, the *feature selector* and the *learning algorithm* (in dynamic feature selection, these two components correspond to the *explainer E* and the *classifier C*, respectively). The typology consists of three classes of methods:

- **Wrapper methods**, in the wording of Guyon and Elisseeff (2003), "utilize the learning machine of interest as a black box to score subsets of variable according to their predictive power." In static feature selection this means greedily searching over subsets of features, training a model with each candidate subset. In the dynamic feature selection world, this is somewhat reminiscent of the ablative analysis of Serrano and Smith (2019) or the leave-one-out method of Li et al. (2016b). Another example is LIME (Ribeiro et al., 2016), which involves repeatedly querying the classifier to label new examples. This class of methods requires multiple calls to the learning algorithm (in the case of static fea-

	Static selection (model interpretability)	**Dynamic selection** (prediction explainability)
Wrappers	Forward selection, backward elimination (Kohavi and John, 1997)	Input reduction (Feng et al., 2018), representation erasure (leave-one-out) (Li et al., 2016b; Serrano and Smith, 2019), LIME (Ribeiro et al., 2016)
Filters	Pointwise mutual information (Church and Hanks, 1990), recursive feature elimination (Guyon et al., 2002)	Input gradient (Li et al., 2016a), layerwise relevance propagation (Bach et al., 2015), thresholding softmax attention
Embedded	ℓ_1-regularization (Tibshirani, 1996), elastic net (Zou and Hastie, 2005)	Stochastic attention (Xu et al., 2015; Lei et al., 2016), sparse attention (**this paper**, §3),

Table 1: Overview of static and dynamic feature selection techniques.

ture selection) or the classifier C (in the case of dynamic feature selection).

- **Filter methods** decide to include/exclude a feature based on some importance metric (such as feature counts or pairwise mutual information). This can be done as a preprocessing step or by training the model once and thresholding the feature weights. In dynamic feature selection, this is done when we examine the gradient of the prediction with respect to each input feature, and then select the features whose gradient has large magnitude.[1] This is the approach taken by the input gradient and layerwise relevance propagation techniques (Li et al., 2016a; Arras et al., 2016; Jain and Wallace, 2019), where this gradient is taken as a measure of feature "importance." Another example is when thresholding softmax attention scores to select relevant input features, as considered by Jain and Wallace (2019) and Wiegreffe and Pinter (2019) in their analyses. This class of methods involve at most one call to the classifier C.

- **Embedded methods**, in traditional feature selection, embed feature selection within the learning algorithm by using a sparse regularizer, such like the ℓ_1-norm (Tibshirani, 1996). Features that receive zero weight become irrelevant and can be removed from the model. In dynamic feature selection, this encompasses methods where the explainer E is directly embedded in the classifier C, such as the method proposed by Lei et al. (2016). We propose in this paper (see §3) an alternative approach via **sparse attention** (Martins and Astudillo, 2016; Peters et al., 2019), where C has a built-in mechanism, resembling ℓ_1-regularization, that allows to *select* relevant input features and assign zero weight to everything else.

---

[1] In linear models this gradient equals the feature's weight.

In §4, we will frame each of the cases above as a component in a communication process, where the explainer E aims to communicate a short message with the relevant features that triggered the classifier C's decisions to a layperson L. The three cases above are distinguished by the way C and E interact.

3 Embedded Sparse Attention

The case where the explainer E is embedded in the classifier C naturally favors faithfulness, since the mechanism that explains the decision (the *why*) can also influence it (the *how*).

Attention mechanisms (Bahdanau et al., 2015) allow visualizing relevant input features that contributed to the model's decision. Attention weights are usually computed by applying a softmax transformation to a vector of scores $\mathbf{s} \in \mathbb{R}^n$, which are themselves a function of a query and key vectors. However, softmax-based attention is *dense*, i.e., it places *some* probability mass to every feature, even if small. The typical approach is to select the top-k words with largest attention weights as the explanation. In the world of static feature selection, this is similar to what happens when using ℓ_2-regularization for filtering: features whose weights have small magnitude are good candidates to remove from the model, i.e., their weights can be truncated to zero.

An alternative is to embed in the classifier an attention mechanism that is inherently **selective**, i.e., which can produce sparse attention distributions, where some input features receive exactly zero attention. Examples are the **sparsemax** (Martins and Astudillo, 2016) and the recently proposed **1.5-entmax** transformation (Peters et al., 2019), described in detail in §A (supplemental material). These sparse attention transformations have been applied successfully in machine translation and morphological inflection applications (Peters et al.,

2019; Correia et al., 2019). We can interpret the words that receive non-zero attention probability as words that have been *selected* to be part of the explanation. This is an embedded method akin of the use of ℓ_1-regularization in static feature selection. We experiment with these sparse attention mechanisms in §5.

4 Explainability as Communication

We now have the necessary ingredients to describe our unified framework for comparing and designing explanation strategies, illustrated in Figure 1.

Our fundamental assumption is that explainability is intimately linked to the ability of an explainer to **communicate** the rationale of a decision in terms that can be understood by a human; the success of this communication is the criterion that dictates how good the explanation is.

4.1 The Classifier-Explainer-Layperson setup

Our framework draws inspiration on Lewis' signaling games (Lewis, 1969) and on the recent work on (emergent) communication (Foerster et al., 2016; Havrylov and Titov, 2017). Our starting point is the classifier $C : \mathcal{X} \to \mathcal{Y}$ which, when given an input $x \in \mathcal{X}$, produces a prediction $\hat{y} \in \mathcal{Y}$. This is the prediction that we want to explain. An explanation is a **message** $m \in \mathcal{M}$, for a predefined message space $\mathcal{M}$ (examples later). We say that the predictions made by C are **explainable** if there is an explainer E that can compose and **successfully communicate** messages m to a layperson L. The success of the communication is dictated by the ability of L to reconstruct $\hat{y}$ with high accuracy from m. In this paper, we experiment with E and L being either humans or machines. In the former case, our framework is inspired by human-grounded evaluation through forward simulation/prediction, as proposed by Doshi-Velez and Kim (2017, §3.2). More formally,

- The **classifier** C is a model for whose decisions we want explanations. For given inputs x, C produces $\hat{y}$ that are hopefully close to the groundtruth y. We are agnostic about the kind of model used as a classifier, but we assume that it might compute certain internal representations h that can expose to the explainer.

- The **explainer** E is a model responsible for producing explanations for the classifier's decisions. It receives the inputs x, the classifier predictions

$\hat{y} = C(x)$, and it has the possibility to access the internal representations h exposed by C. It outputs messages $m \in \mathcal{M}$ that can be seen as a "rationale" for $\hat{y}$. The message $m = E(x, \hat{y}, h)$ should be simple and compact enough to be easily transmitted and understood by the layperson L. In this paper, we constrain the messages $m \in \mathcal{M}$ to be bags-of-words (BoWs) extracted from the textual input x, up to a maximum length of k words.[2]

- The **layperson** L is a simple model (e.g., a linear classifier) that receives the message m as input, and predicts a final output $\tilde{y} = L(m)$. The communication is successful if $\tilde{y} = \hat{y}$. Given a test set $x_1, \ldots, x_N$, we evaluate the **communication success rate** as the fraction of examples for which the communication was successful:

$$\text{CSR} = \frac{1}{N} \sum_{n=1}^{N} \left[\left[C(x_n) = L(E(x_n, C(x_n))) \right] \right],$$

(1)

where $[[\cdot]]$ is the Iverson bracket notation.

Under this framework, we regard the communication success rate (CSR) as a quantifiable measure of explainability: a high CSR means that the layperson L is able to replicate the classifier C's decisions a large fraction of the time when presented with the messages given by the explainer E; this certifies that E's messages are informative enough.

4.2 Task examples

We show examples of how we can model the explainer and layperson for different NLP problems.

Text classification. We let $x \in \mathcal{X}$ be a document (sequence of words), and the output space $\mathcal{Y}$ a set of labels (e.g. topics or sentiment labels). If C is an attention based classifier, then E could just be a deterministic model that takes the attention probabilities from C and truncates them to create a message m as a small BoW.

Natural language entailment. Here, x is a pair of texts (a premise and an hypothesis) and the labels in $\mathcal{Y}$ are entailment, contradiction, and neutral. We let messages be again BoWs, and we constrain them to be selected from the premise (and concatenated with the full hypothesis), as shown in Fig. 2.

---

[2]Note that our framework is flexible about the choice of this message space $\mathcal{M}$. For example, explanations could also be *prototypes*, i.e., small subsets of training examples.
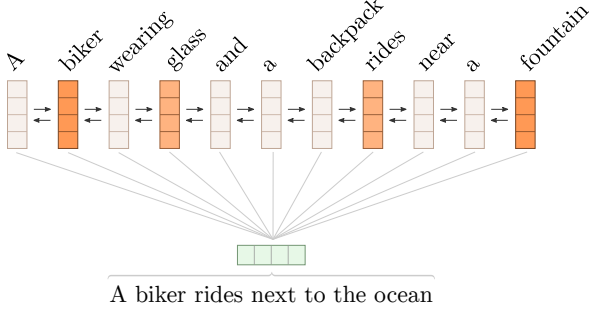
Figure 2: Example of sparse attention for natural language entailment. The selected premise words ("biker", "glass", "rides", and "fountain") form the message, together with the hypothesis.
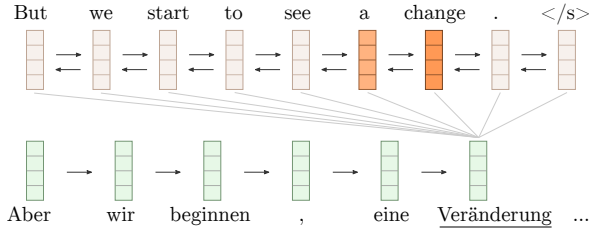

Figure 3: Example of sparse attention for machine translation. When the model is translating the word "Veränderung", the source words "a" and "change" are treated as explanation and sent as message.

Machine translation. We consider the decision taken by a neural machine translation (NMT) system about the tth target word to generate (y), given the source sentence x and the previously generated words $y_{1:t-1}$. Note that in this example $\mathcal{Y}$ is large (the entire target vocabulary). The message is the concatenation of k source words (ranked by importance, without any word order information) with the prefix $y_{1:t-1}$. The layperson must predict the target word given this limited information. Fig. 3 illustrates how attention can be treated as explanation in the context of MT.

4.3 Embedded and post-hoc explanations

Our framework is flexible, permitting different configurations for the C, E, and L. In the embedded approaches described in §2, the explainer E can be embedded as a component of the classifier C, e.g., as a sparse attention mechanism (as described in §3). On the other hand, in wrappers and filters the two components can be decoupled, and E can query C for new examples or request gradient information. In this case, E is a *post-hoc explainer* which provides model-agnostic explanations, treating C as a black box.

4.4 Joint training of explainer and layperson

Until now, we have assumed that the explainer E is given beforehand, and that the layperson L is a simple model that needs to be trained to assess the explanatory ability of E. But what if E is also trained? For communication to succeed, E and L have to agree on a protocol that ensures informative messages. We will now see how E and L can be jointly trained, by learning to play a cooperative game (Lewis, 1969). Special care needs to be taken to ensure that the protocol they agree upon is not a "trivial" one: it has to retain information that is indeed useful to be regarded as a plausible, potentially faithful, explanation. We propose a strategy to ensure this, which will be validated using human evaluation in §6. We assume in this setting that explanations are extracted after C is trained, i.e., E is a post-hoc explainer.

A good explanation must be *informative* (both about the decision and the model's decision process), *compact*, and *understandable*. The last two requirements are ensured by defining $\mathcal{M}$ properly. But how to ensure the first requirement? To fully model the communication process and maximize the informativeness of the explanations m, we consider a strategy where both the explainer E_θ and layperson L_ϕ are **trained models** (with parameters θ and ϕ), learned together to optimize a multi-task objective with two terms:

- A **reconstruction term** that controls the information about the classifier's decision $\hat{y}$. We use a cross-entropy loss on the output of the layperson L, using $\hat{y}$ (and not the true label y) as the groundtruth: $\mathcal{L}(\phi, \theta) = -\log p_\phi(\hat{y} \mid m)$, where m is the output of the explainer E_θ.

- A **faithfulness term** that encourages the explainer E to take into account, to some degree, the classifier's decision process when producing its explanation m. To do this, we add a reconstruction loss term (using the cosine dissimilarity) that forces the explainer to predict also a continuous vector $\tilde{h}$ that should match the internal representation h that the classifier C exposes: $\Omega(\theta) = 1 - \cos(\tilde{h}(E_\theta), h)$.

The objective function to minimize is a combination of these two terms, $\mathcal{L}_\Omega(\phi, \theta) := \lambda \Omega(\theta) + \mathcal{L}(\phi, \theta)$, where $\lambda \geq 0$ is a constant tuned in a validation set. This function is minimized in a training set that contains pairs $(x, \hat{y})$. Therefore, in this

Name	# Train	# Test	Avg. tokens	# Classes
SST	6920	1821	19	2
IMDB	25K	25K	280	2
AgNews	115K	20K	38	2
Yelp	5.6M	1M	130	5
SNLI	549K	9824	14 / 8	3
IWSLT	206K	2271	20 / 18	134,086

Table 2: Dataset statistics. The average number of tokens for SNLI is related to the premise and hypothesis, and for IWSLT to the source and target sentences.

model the message m is latent and works as a "bottleneck" for the layperson L, which does not have access to the full input x, to guess the classifier's prediction $\hat{y}$—similar models have been devised in the context of emergent communication (Lazaridou et al., 2016; Foerster et al., 2016; Havrylov and Titov, 2017) and sparse auto-encoders (Trifonov et al., 2018; Subramanian et al., 2018). To optimize the loss above with gradient backpropagation and avoid the need for a non-differentiable operation such as argmax or top-k, at training time we use sparsemax attention (which is end-to-end differentiable) to select the relevant words in the message. One important concern in this model is to avoid E and L to agree on a trivial protocol to maximize the communication accuracy. To prevent this, we set a linear schedule for the probability of the explainer accessing the predictions of the classifier ($\hat{y}$). At the end of training, the explainer will access it with probability β (e.g. 50%).

5 Experiments

We experimented with our framework in three NLP tasks (text classification, entailment, and machine translation), using the setup in §4.2.

5.1 Text classification

We used 4 datasets (SST, IMDB, AgNews, Yelp), with statistics in Table 2. We picked the same datasets as Jain and Wallace (2019) and Wiegreffe and Pinter (2019), excluding the smallest ones. For SST and IMDB we used the standard splits, and for AgNews and Yelp we randomly split the dataset, leaving 85% for training and 15% for test.

Classifier C. Each input word x_i is mapped to 300D-pretrained GloVe embeddings (Pennington et al., 2014), kept frozen, followed by a 128D-bidirectional LSTM layer (BiLSTM), resulting in vectors $\mathbf{h}_1, \ldots, \mathbf{h}_n$. We score each of these vectors using the additive formulation from Bahdanau

et al. (2015), applying an attention transformation to convert the resulting scores $\mathbf{s} \in \mathbb{R}^n$ to a probability distribution $\pi \in \triangle^n$. We use this to compute a contextual vector $\mathbf{c} = \sum_{i=1}^{n} \pi_i \mathbf{h_i}$, which is fed into the output softmax layer that predicts $\hat{y}$. We trained our classifiers for at most 10 epochs, with a patience of 5 (for Yelp, we used 5 and 3, respectively), using AdamW (Loshchilov and Hutter, 2019), ℓ_2 regularization with $\lambda = 0.01$, and default hyperparameters from PyTorch.

Layperson L and explainer E. We used a simple linear BoW model as the layperson L. We evaluated three different types of explainers: (i) **Erasure**, a wrapper similar to the leave-one-out approaches of Jain and Wallace (2019) and Serrano and Smith (2019). We obtain the word with largest attention, "zero out" its input vector, and repass the whole input with the erased vector to the classifier C. We produce the message by repeating this procedure k times and storing the erased words; (ii) **Top-k gradients**, a filter approach that ranks word importance by their "input × gradient" product, $|\frac{\partial \hat{y}}{\partial \mathbf{x}_i} \cdot \mathbf{x}_i|$ (Ancona et al., 2018; Wiegreffe and Pinter, 2019). The top-k words are selected as the message; (iii) **Top-k and selective attention:** We experimented both using attention as a *filter*, by selecting the top-k most attended words as the message, and *embedded* in the classifier C, by using the selective attentions described in §3 (1.5-entmax and sparsemax). We also report a **random** baseline, which randomly picks k words as the message.

Table 3 (columns 2–6) reports results for the communication success rate (CSR, Eq. 1) and for the accuracy of the layperson (ACC). By looking at the ACC column, we see a consistent drop from the RNN classifier to the layperson, regardless of the explainer. This is expected, since the layperson is a much weaker BoW classifier, and only has access to a limited number of words in the document. Note, however, that for some explainers (all the attention-based ones), this layperson outperforms a BoW classifier with access to all words. This is reassuring, as it shows that *the layperson guided by the explainer outperforms an unguided layperson.*

Regarding the CSR metric, the first thing to note is that, as expected, the random baseline does much worse than the other explainers, for all datasets. Among the non-trivial explainers, the erasure and top-k attention explainers have similar performances in terms of CSR, with a slight advantage for attention methods. Note that the attention

EXPLAINER	SST		IMDB		AGNEWS		YELP		SNLI	
	CSR	ACC	CSR	ACC	CSR	ACC	CSR	ACC	CSR	ACC
BoW	-	82.54	-	88.96	-	95.62	-	68.78	-	69.81
RNN	-	**86.16**	-	**91.79**	-	**96.28**	-	**75.80**	-	**78.34**
Random	69.41	70.07	67.30	66.67	92.38	91.14	58.27	53.06	75.83	68.74
Erasure	80.12	81.22	92.17	88.72	97.31	95.41	78.72	68.90	77.88	70.04
Top-k gradient	79.35	79.24	86.30	83.93	96.49	94.86	70.54	62.86	76.74	69.40
Top-k softmax	84.18	82.43	93.06	89.46	**97.59**	95.61	81.00	70.18	78.66	71.00
Top-k 1.5-entmax	**85.23**	**83.31**	93.32	89.60	97.29	**95.67**	82.20	70.78	80.23	73.39
Top-k sparsemax	**85.23**	81.93	**93.34**	89.57	95.92	94.48	82.50	70.99	**82.89**	**74.76**
Selec. 1.5-entmax	83.96	82.15	92.55	**89.96**	97.30	95.66	81.38	70.41	77.25	71.44
Selec. sparsemax	**85.23**	81.93	93.24	89.66	95.92	94.48	**83.55**	**71.60**	82.04	73.46

Table 3: Results for text classification and SNLI datasets. CSR stands for the communication success rate, and ACC is the accuracy of the classifier/layperson. The top rows are the original classifiers, which access all words. The middle rows report performance for random, wrapper and filter explainers, for fixed k-word messages (the values of k for the several datasets are $\{5, 10, 10, 10, 4\}$, respectively). The bottom rows correspond to embedded methods where k is given automatically via sparsity. The average k obtained by 1.5-entmax and sparsemax are: SST: $\{4.65, 2.59\}$; IMDB: $\{28.23, 12.94\}$; AGNEWS $\{5.65, 4.14\}$; YELP: $\{60.61, 23.86\}$; SNLI: $\{12.96, 8.27\}$.

explainers have the important advantage, though, of requiring a single call to the classifier, whereas the erasure methods, being wrappers, require k calls. On the other hand, the top-k gradient method had much worse performance (except for the AGNEWS dataset), suggesting that the words that make the label scores vary the most locally are not necessarily the most informative ones.[3] Regarding the different attention models, we see that, excepting for AG-NEWS, sparse transformations tend to have slightly better CSR and ACC.[4] The embedded (sparse) attention methods achieved communication scores in par with the top-k attention methods without a prescribed k. We analyze the trade-off between message length and CSR in §5.4, and we show examples of messages for the several systems in §A.2.

5.2 Natural language entailment

We conducted experiments on the SNLI dataset (Table 2). We used a similar classifier C as the one in §5.1, except that now we have two inputs (the premise and the hypothesis). We used two independent BiLSTM layers, one for each. We used the

additive attention of Bahdanau et al. (2015) with the last hidden state of the hypothesis as the query and the premise vectors as keys. The layperson differs from the one in §5.1 as follows: it uses a BiLSTM to encode the hypothesis, and then the BoWs from the explainer is passed through a linear projection and summed with the last hidden state of the BiLSTM, before the output layer.

The rightmost column of Table 3 shows a comparison for the explainers introduced in §5.1. The conclusion for SNLI is essentially the same as in text classification. The random baseline is very competitive, probably because just looking at the hypothesis is already a strong baseline (Gururangan et al., 2018). We also see that sparse attentions consistently perform better than other explainers. Moreover, when we truncate the attention distribution to top-k words the results are better then their fully embedded counterparts. We investigate the impact of the hyperparameter k in §5.4.

5.3 Machine Translation

We used the EN→DE IWSLT 2017 dataset (Cettolo et al., 2017), with the standard splits (Table 2), using the JoeyNMT framework (Kreutzer et al., 2019). We employed beam search decoding with beam size of 5, achieving a BLEU score of 20.49, 21.12 and 20.75 for softmax, 1.5-entmax and sparsemax, respectively. The layperson is a model that uses an unidirectional 256D LSTM to encode the translation prefix, and a feed-forward layer to encode the concatenation of k source word embeddings (the message) to a 256D vector. The

---

[3]A potential reason for this is the way the model makes its decisions. Attention has a direct influence on the decisions, because it is an inside component during the learning process. Gradients and erasure, however, are extracted once the learning process is finished. The reason might be similar to filter methods being generally inferior to embedded methods in static feature selection, since they ignore feature interactions that may jointly play a role in model's decisions.

[4]For AGNEWS, we found that simply selecting the first k words of the text yields very high CSR: 93.92 and ACC: 92.49. This is consistent with previous findings by Wiegreffe and Pinter (2019).

EXPLAINER	$k=0$	$k=1$	$k=3$	$k=5$
Top-k gradient	21.99	35.21	38.33	40.30
Top-k softmax	21.99	**62.58**	62.82	62.64
Top-k 1.5-entmax	**22.31**	62.53	**63.48**	**62.69**
Top-k sparsemax	22.14	62.21	61.94	61.92

Table 4: Results for IWSLT. Reported are CSR scores.

two vectors are concatenated and passed to a linear output layer to predict the next word $\tilde{y} \in \mathcal{Y}$ (see Fig. 3).

Results comparing different filtering methods varying k are shown in Table 4. We show the CSR as we varied $k \in \{0, 1, 3, 5\}$. Again, top-k attention performed better than top-k gradient, in this case with a wider margin. In general, all methods perform better as we increase k, but we can already see a degradation of performance when $k = 5$ for all attention based explainers. An interesting case is when $k = 0$, meaning that L has no access to the source sentence, behaving like an unconditioned language model. In this case the performance is much worse, indicating that both explainers are selecting relevant tokens when $k > 0$.

5.4 Trade-off between k and CSR.

Figure 4 shows the trade-off between the length of the message and the communication success rate for different values of k both for IMDB (text classification) and SNLI (natural language inference).

We see that as k increases, CSR starts by increasing but then it starts dropping when k becomes too large. This matches our intuition: in the two extreme cases where $k = 0$ and where k is the document length (corresponding to a full bag-of-words classifier) the message has no information about how the classifier C behaves. This behaviour is more clear for SNLI, where the average length of the premise is of 15 words only. By setting $k = 0$, meaning that the layperson L only looks at the hypothesis, the CSR is reasonably high ($\sim$74%), but soon as we include a single word in the message this baseline is surpassed by 4 points or more. This is consistent with the previous finding that by considering only the hypothesis it is possible to achieve a high accuracy for SNLI (Gururangan et al., 2018).

6 Human Evaluation

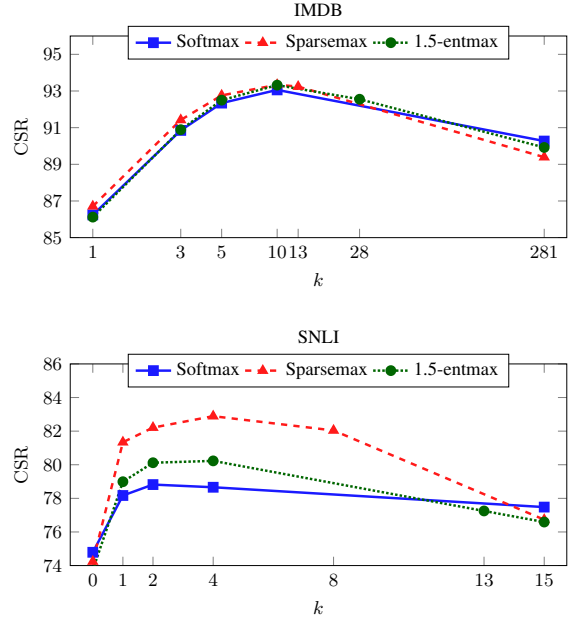**Joint training of E and L.** So far we compared several given explainers, but what happens if we

Figure 4: Message sparsity analysis for IMDB (left) and SNLI (right). For SNLI, $k = 0$ corresponds to a case where the layperson only sees the hypothesis. The rightmost entry is the average length of the examples in the test set, and therefore it represents an explainer that simply pass forward all words to the layperson (i.e. a full bag-of-words). The average k for sparsemax and 1.5-entmax are, respectively: 13 and 28 for IMDB; 8 and 13 for SNLI.

train E and L jointly to optimize CSR directly, as described in §4.4? We experiment with the IMDB and SNLI datasets, with $\lambda = 2$ for IMDB, $\lambda = 20$ for SNLI, and $\beta = 20\%$ for both. We contrast this with using humans for either the layperson, the explainer, or both.

Human layperson. We carried a portion of the explainers used in the previous experiments and we randomly selected 200 documents for IMDB and 100 for SNLI to be annotated by humans. The explanations (i.e. the selected words) were shuffled and displayed as a cloud of words to two annotators, who had to decide the label of each document. For SNLI, we show the entire hypothesis as raw text and the premise as a cloud of words. The annotation interface is shown in Fig. 5 (§A.3).

Human explainer. We also consider explanations generated by humans rather than machines. To this end, we used the e-SNLI corpus (Camburu et al., 2018), which extends the SNLI with human labeled marks on relevant words. Unfortunately, the e-SNLI corpus does not provide highlights over the premise for neutral pairs, and therefore we re-

EXPLAINER	IMDB				SNLI			
	H-CSR	L-CSR	H-ACC	L-ACC	H-CSR	L-CSR	H-ACC	L-ACC
Top-k gradient	73.75	84.50	70.75	80.50	66.00	76.00	78.50	75.00
Top-k softmax	88.50	93.00	87.00	88.00	70.50	75.00	81.00	74.00
Top-k 1.5-entmax	87.50	92.50	85.00	86.50	**74.50**	80.00	85.00	76.00
Top-k sparsemax	90.00	89.50	**89.50**	88.00	73.00	86.00	81.50	76.00
Joint E and L	**93.25**	**99.50**	87.75	**92.50**	72.00	**99.00**	78.00	**80.00**
Human highlights	-	-	-	-	69.00	76.00	**86.50**	77.00

Table 5: Results for human evaluation. H-CSR/ACC stands for human annotator (layperson) performance, L-CSR/ACC stand for machine layperson performance.

moved the neutral class from the test set and obtained explanations using our trained models. We summarize our results in Table 5.

As in our previous experiments, better results were found both in terms of CSR and ACC for top-k attention methods in comparison to top-k gradient. Although we reported both H-CSR and L-CSR for the human highlights explainer, we point out that this explainer is not trained to match the classifier's C decisions, which may explain the relatively low scores. The ACC of top-k attention models and human highlights explainers are close, reinforcing again the good results for attention-based models. Moreover, among the different attention explainers, we can see that sparsemax and 1.5-entmax outperforms softmax in human scores.

As expected, the joint explainer achieves very high L-CSR, also outperforming all explainers in terms of L-ACC. The joint explainer was also able to achieve good human performance on IMDB, showing that it was able to optimize the communication and yet produce informative explanations. Whereas for SNLI, the human metrics are a bit lower in comparison with attention-based explainers, suggesting that in this case its explanations were not so informative. Outputs for these explainers can be consulted in §A.2.

7 Related Work

There is a body of recent work related to analysis and interpretation of neural networks. Our work is about *prediction explainability*, not to be confused with transparency or model interpretability—the difference between these concepts is made precise by Doshi-Velez and Kim (2017); Lipton (2018); Gilpin et al. (2018); Rudin (2019).

Recent works questioned the interpretative ability of attention mechanisms and whether they can be regarded as a form of explanation or not (Jain and Wallace, 2019; Serrano and Smith, 2019).

Wiegreffe and Pinter (2019) distinguished between faithful and plausible explanations and introduced several diagnostic tools to assess the usefulness of attention for explainability. None of these works, however, considers the sparse selective attention mechanisms we propose in this paper. Our goal is not to distinguish between faithful and plausible explanations. In contrast, we focus on better characterizing when explanations should be considered plausible, by providing a framework that gives an objective answer to this question. As discussed in Wiegreffe and Pinter (2019, §5), plausible explanations are very important even if not faithful: Rudin (2019) defines explainability as a plausible reconstruction of the decision-making process, and Riedl (2019) argues that they mimic what humans do when rationalizing past actions.

Gradient methods, such as DeepLIFT and Integrated Gradients (Bach et al., 2015; Montavon et al., 2018), leave-one-out strategies (Feng et al., 2018; Serrano and Smith, 2019), or post-hoc explainers such as LIME and Influence Functions (Ribeiro et al., 2016; Koh and Liang, 2017), have been proposed to generate or diagnose explanations, but the link between these approaches and filters and wrappers in the feature selection literature has not been made before. We believe the connections revealed in §2 will be potentially useful to develop new methods for model explainability.

Our framework draws inspiration from the field of emergent communication (Lazaridou et al., 2016; Foerster et al., 2016; Havrylov and Titov, 2017); some of the ideas proposed here (such as making the system end-to-end differentiable through sparsemax) may also be relevant in that field. The sparse, selective attention mechanisms that we propose in §3 are inspired by the "explanation selection" principle articulated by Miller (2019, §4): "Similar to causal connection, people do not typically provide all causes for an event

as an explanation. Instead, they select what they believe are the most relevant causes." Sparse and stochastic attention have been considered in several prior works (Xu et al., 2015; Lei et al., 2016; Martins and Astudillo, 2016; Peters et al., 2019; Bastings et al., 2019), but a systematic comparison with other attention and explanation strategies was still missing. Other approaches include Lei et al. (2016); Alvarez-Melis and Jaakkola (2017); Bastings et al. (2019).

Our work is also related to literature on sparse auto-encoders, which seek sparse overcomplete vector representations to improve model interpretability (Faruqui et al., 2015; Trifonov et al., 2018; Subramanian et al., 2018). In contrast to these works, we consider the probability distribution induced by sparse attention mechanisms as a form of explanation. Mullenbach et al. (2018) also use human evaluation to show that attention mechanism identifies meaningful explanations, which is consistent with our findings in §6.

Concurrently to our work, DeYoung et al. (2019) recently proposed a toolkit to compare different explanatory methods, distinguishing between comprehensive and sufficient rationales. Our work focus on sufficient rationales, not necessarily comprehensive, following the selective explanation principle of Miller (2019) stated above. Yet, their approach is orthogonal to ours and we believe both can benefit from each other.

8 Conclusions

We proposed a unified framework that regards explainability as a communication problem between an explainer and a layperson about a classifier's decision. In doing so, we organized existing approaches in a typology that makes a bridge between traditional feature selection and modern explanation techniques, and which motivates our newly proposed embedded methods based on selective attention. In our experiments in text classification, entailment, and machine translation, we observed that attention mechanisms tend to outperform gradient methods and representation erasure on communication success rate, using both machines and humans as the layperson.

References

David Alvarez-Melis and Tommi Jaakkola. 2017. A causal framework for explaining the predictions of black-box sequence-to-sequence models. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 412–421.

Marco Ancona, Enea Ceolini, Cengiz Öztireli, and Markus Gross. 2018. Towards better understanding of gradient-based attribution methods for deep neural networks. In *Proc. ICLR*.

Leila Arras, Franziska Horn, Grégoire Montavon, Klaus-Robert Müller, and Wojciech Samek. 2016. Explaining predictions of non-linear classifiers in NLP. In *Proc. RepL4NLP*, pages 1–7.

Leila Arras, Grégoire Montavon, Klaus-Robert Müller, and Wojciech Samek. 2017. Explaining recurrent neural network predictions in sentiment analysis. In *Proc. WASSA*.

Sebastian Bach, Alexander Binder, Grégoire Montavon, Frederick Klauschen, Klaus-Robert Müller, and Wojciech Samek. 2015. On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. *PloS one*, 10(7):130–140.

Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. In *Proc. ICLR*.

Joost Bastings, Wilker Aziz, and Ivan Titov. 2019. Interpretable neural predictions with differentiable binary variables. In *Proc. ACL*.

Oana-Maria Camburu, Tim Rocktäschel, Thomas Lukasiewicz, and Phil Blunsom. 2018. e-snli: Natural language inference with natural language explanations. In *Proc. NeurIPS*, pages 9539–9549.

Mauro Cettolo, Marcello Federico, Luisa Bentivogli, Niehues Jan, Stüker Sebastian, Sudoh Katsuitho, Yoshino Koichiro, and Federmann Christian. 2017. Overview of the iwslt 2017 evaluation campaign. In *Proc. IWSLT*, pages 2–14.

Kenneth Ward Church and Patrick Hanks. 1990. Word association norms, mutual information, and lexicography. *Computational linguistics*, 16(1):22–29.

Gonçalo M Correia, Vlad Niculae, and André FT Martins. 2019. Adaptively sparse transformers. In *Proc. EMNLP-IJCNLP*, pages 2174–2184.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proc. NAACL-HLT*, pages 4171–4186.

Jay DeYoung, Sarthak Jain, Nazneen Fatema Rajani, Eric Lehman, Caiming Xiong, Richard Socher, and Byron C Wallace. 2019. Eraser: A benchmark to evaluate rationalized nlp models. *arXiv preprint arXiv:1911.03429*.

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

Manaal Faruqui, Yulia Tsvetkov, Dani Yogatama, Chris Dyer, and Noah A. Smith. 2015. Sparse overcomplete word vector representations. In *Proc. IJCNLP*, pages 1491–1500.

Shi Feng, Eric Wallace, Alvin Grissom II, Mohit Iyyer, Pedro Rodriguez, and Jordan Boyd-Graber. 2018. Pathologies of neural models make interpretations difficult. In *Proc. EMNLP*, pages 3719–3728.

Jakob Foerster, Ioannis Alexandros Assael, Nando De Freitas, and Shimon Whiteson. 2016. Learning to communicate with deep multi-agent reinforcement learning. In *Advances in neural information processing systems*, pages 2137–2145.

Leilani H Gilpin, David Bau, Ben Z Yuan, Ayesha Bajwa, Michael Specter, and Lalana Kagal. 2018. Explaining explanations: An overview of interpretability of machine learning. In *Proc. DSAA*, pages 80–89.

Yoav Goldberg. 2017. Neural network methods for natural language processing. *Synthesis Lectures on Human Language Technologies*, 10(1):1–309.

Suchin Gururangan, Swabha wayamdipta, Omer Levy, Roy Schwartz, Samuel Bowman, and Noah A. Smith. 2018. Annotation artifacts in natural language inference data. In *Proc. NAACL-HLT*.

Isabelle Guyon and André Elisseeff. 2003. An introduction to variable and feature selection. *Journal of machine learning research*, 3(Mar):1157–1182.

Isabelle Guyon, Jason Weston, Stephen Barnhill, and Vladimir Vapnik. 2002. Gene selection for cancer classification using support vector machines. *Machine learning*, 46(1-3):389–422.

Serhii Havrylov and Ivan Titov. 2017. Emergence of language with multi-agent games: Learning to communicate with sequences of symbols. In *Proc. NeurIPS*, pages 2149–2159.

Sarthak Jain and Byron C Wallace. 2019. Attention is not explanation. In *Proc. NAACL-HLT*.

Pang Wei Koh and Percy Liang. 2017. Understanding black-box predictions via influence functions. In *Proc. ICML*.

Ron Kohavi and George H John. 1997. Wrappers for feature subset selection. *Artificial intelligence*, 97(1-2):273–324.

Julia Kreutzer, Joost Bastings, and Stefan Riezler. 2019. Joey nmt: A minimalist nmt toolkit for novices. In *Proc. of EMNLP-IJCNLP: System Demonstrations*, pages 109–114.

Angeliki Lazaridou, Alexander Peysakhovich, and Marco Baroni. 2016. Multi-agent cooperation and the emergence of (natural) language. In *Proc. ICLR*.

Tao Lei, Regina Barzilay, and Tommi Jaakkola. 2016. Rationalizing neural predictions. In *Proc. EMNLP*, pages 107–117.

David Lewis. 1969. Convention: A philosophical study.

Jiwei Li, Xinlei Chen, Eduard Hovy, and Dan Jurafsky. 2016a. Visualizing and understanding neural models in nlp. In *Proc. NAACL-HLT*, pages 681–691.

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

Zachary C. Lipton. 2018. The mythos of model interpretability. *Commun. ACM*, 61(10):36–43.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *Proc. ICLR*.

Andre Martins and Ramon Astudillo. 2016. From softmax to sparsemax: A sparse model of attention and multi-label classification. In *Proc. ICML*, pages 1614–1623.

Tim Miller. 2019. Explanation in artificial intelligence: Insights from the social sciences. *Artificial Intelligence*, 267:1–38.

Grégoire Montavon, Wojciech Samek, and Klaus-Robert Müller. 2018. Methods for interpreting and understanding deep neural networks. *Digital Signal Processing*, pages 1–15.

James Mullenbach, Sarah Wiegreffe, Jon Duke, Jimeng Sun, and Jacob Eisenstein. 2018. Explainable prediction of medical codes from clinical text. In *Proc. NAACL-HLT*, pages 1101–1111.

Jeffrey Pennington, Richard Socher, and Christopher D. Manning. 2014. Glove: Global vectors for word representation. In *Proc. EMNLP*, pages 1532–1543.

Ben Peters, Vlad Niculae, and André FT Martins. 2019. Sparse sequence-to-sequence models. *Proc. ACL*.

Matthew E Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *Proceedings of NAACL-HLT*, pages 2227–2237.

Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. 2016. Why should i trust you?: Explaining the predictions of any classifier. In *Proc. ACM SIGKDD*, pages 1135–1144. ACM.

Mark O Riedl. 2019. Human-centered artificial intelligence and machine learning. *Human Behavior and Emerging Technologies*, 1(1):33–36.

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.

Sofia Serrano and Noah A Smith. 2019. Is attention interpretable? In *Proc. ACL.*

Anant Subramanian, Danish Pruthi, Harsh Jhamtani, Taylor Berg-Kirkpatrick, and Eduard Hovy. 2018. Spine: Sparse interpretable neural embeddings. In *Thirty-Second AAAI Conference on Artificial Intelligence.*

Robert Tibshirani. 1996. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society*, 58(1):267–288.

Valentin Trifonov, Octavian-Eugen Ganea, Anna Potapenko, and Thomas Hofmann. 2018. Learning and evaluating sparse interpretable sentence embeddings. In *Proc. EMNLP Workshop BlackboxNLP.*

Constantino Tsallis. 1988. Possible generalization of boltzmann-gibbs statistics. *Journal of Statistical Physics.*

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 *Proc. NeurISP*, pages 5998–6008.

Sarah Wiegreffe and Yuval Pinter. 2019. Attention is not not explanation. In *Proc. EMNLP-IJCNLP.*

Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhudinov, Rich Zemel, and Yoshua Bengio. 2015. Show, attend and tell: Neural image caption generation with visual attention. In *Proc. ICLR*, pages 2048–2057.

Omar F Zaidan and Jason Eisner. 2008. Modeling annotators: A generative approach to learning from annotator rationales. In *Proc. EMNLP*, pages 31–40.

Hui Zou and Trevor Hastie. 2005. Regularization and variable selection via the elastic net. *Journal of the royal statistical society*, 67(2):301–320.

A Supplemental Material

A.1 Sparse attention

A natural way to get a sparse attention distribution is by using the **sparsemax transformation** (Martins and Astudillo, 2016), which computes an Euclidean projection of the score vector onto the probability simplex $\triangle^n := \{\mathbf{p} \in \mathbb{R}^n \mid \mathbf{p} \geq \mathbf{0}, \; \mathbf{1}^\top \mathbf{p} = 1\}$, or, more generally, the α**-entmax transformation** (Peters et al., 2019):

$$\alpha\text{-entmax}(\mathbf{s}) := \arg\max_{\mathbf{p} \in \triangle^n} \mathbf{p}^\top \mathbf{s} + H_\alpha(\mathbf{p}), \tag{2}$$

where H_α is a generalization of the Shannon and Gini entropies proposed by Tsallis (1988), parametrized by a scalar $\alpha \geq 1$:

$$H_\alpha(\mathbf{p}) := \begin{cases} \frac{1}{\alpha(\alpha-1)} \sum_j (p_j - p_j^\alpha), & \alpha \neq 1 \\ -\sum_j p_j \log p_j, & \alpha = 1. \end{cases} \tag{3}$$

Setting $\alpha = 1$ recovers the softmax function, while for any value of $\alpha > 1$ this transformation can return a sparse probability vector. Letting $\alpha = 2$, we recover sparsemax. A popular choice is $\alpha = 1.5$, which has been successfully used in machine translation and morphological inflection applications (Peters et al., 2019).

A.2 Examples of explanations

Table 6 shows the output of erasure, gradient, attention and joint explainers for four examples of IMDB. In Table 7 we also include the human highlights explainer for four pairs of SNLI.

A.3 Annotation interface

Figure 5 shows a snapshot of the annotation interface used for the experiments described in §6.
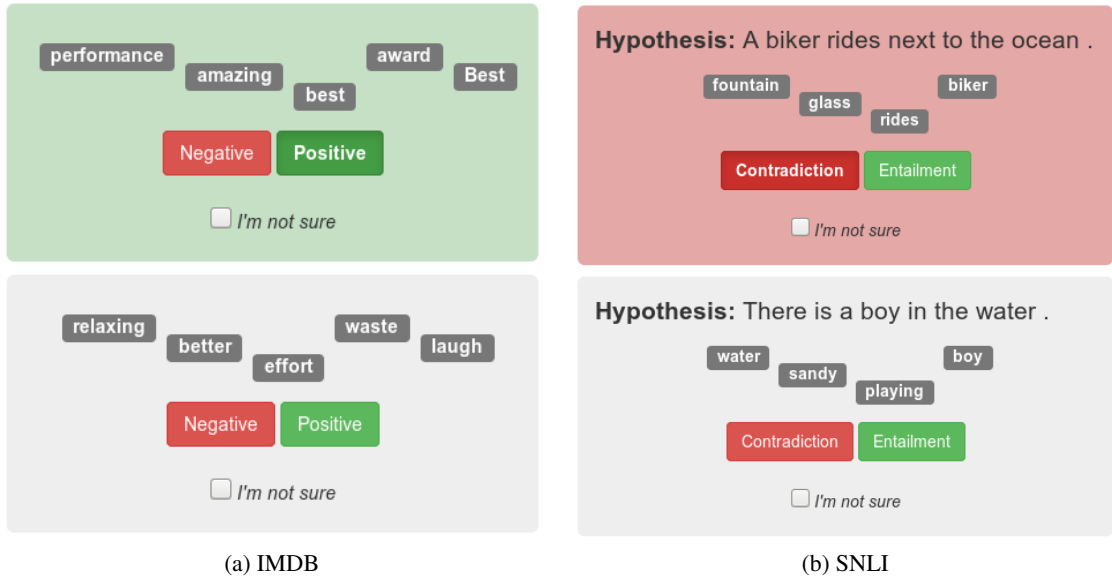

(a) IMDB　　　　　　　　　　　　　　　(b) SNLI

Figure 5: Snapshot of the annotation interface.

EXPLAINER	EXPLANATIONS
	(positive) Fun movie! Great for the kids - they found it very entertaining. Somewhat predictable, but there are a few surprises. Great movie to watch if you're looking for something just to entertain (don't expect to be seeing a classic!)
Erasure	Great Great entertaining ! Fun
Top-k gradient	Great Great ! movie it
Top-k softmax	Great Great movie entertaining !
Top-k 1.5-entmax	Great Great to ! entertaining
Top-k sparsemax	Great Great entertaining ! Fun
Joint E and L	very entertaining Great movie
	(negative) This sequel to Problem Child is just as bad as the first one. It still teaches kids that it's O.K. to be bad. It's impossible for me to recommend this movie to anyone .
Erasure	bad recommend Problem bad impossible
Top-k gradient	recommend Problem this to impossible
Top-k softmax	bad recommend impossible bad Problem
Top-k 1.5-entmax	bad impossible to bad recommend
Top-k sparsemax	bad impossible recommend bad kids
Joint E and L	bad bad
	(positive) When "The Net" was first being advertised, the ads made it look ridiculous. Then, when I saw it, it was actually quite good. Angela Bennett (Sandra Bullock) spends her days working on the computer and has never gotten to know her neighbors. Then, through a series of events, her identity gets erased by a cabal of shadowy people, and she can't prove that she exists. Some parts of the movie are a little bit far-fetched; you'd probably know which parts if you saw the movie. Still, it's a good look into what the existence of the Internet may have wrought on unsuspecting people. I do recommend it.
Erasure	good good recommend ridiculous Still
Top-k gradient	ridiculous . good it recommend
Top-k softmax	good good recommend ridiculous do
Top-k 1.5-entmax	good do good actually quite
Top-k sparsemax	good good Still quite recommend
Joint E and L	good good movie recommend movie
	(negative) The film itself is only a compilation of scenes which have no inherent meaning to someone living outside of Russia. I won't deny that some of the images and techniques were quite revolutionary at the time (filmed 1928) but the problem with the film is that it has no interest to the intellectual or common man. We are merely watching an arranged form of pictures, ranging from a one arm man beating a horse, to a toothless soldier in the war . Everything in between is awkward, haphazard and quite unnecessary. It would have been possible to invent a forum which kept the viewer interested but this would not be it although the method of the director is quite brilliant. In all, one should view this if they are an art student, on hallucinogenic drugs, or a student of pre-Tarkovskian cinema.
Erasure	unnecessary brilliant haphazard compilation Everything
Top-k gradient	brilliant only problem haphazard unnecessary
Top-k softmax	unnecessary brilliant Everything compilation but
Top-k 1.5-entmax	brilliant unnecessary no problem no
Top-k sparsemax	brilliant unnecessary haphazard Everything problem
Joint E and L	no no would have

Table 6: Examples of extracted explanations for IMDB.

EXPLAINER	EXPLANATIONS
	(entailment)
	hypothesis: There is a person outside.
	premise: An adventurous man navigates through the jungle with a long stick.
Erasure	navigates stick jungle man
Top-k gradient	the jungle navigates .
Top-k softmax	navigates jungle stick adventurous
Top-k 1.5-entmax	adventurous man the navigates
Top-k sparsemax	navigates adventurous man the
Joint E and L	. jungle navigates stick
Human highlights	navigates through the jungle
	(contradiction)
	hypothesis: A biker rides next to the ocean.
	premise: A biker wearing glass and a backpack rides near a fountain.
Erasure	fountain glass rides biker
Top-k gradient	fountain near backpack a
Top-k softmax	fountain glass rides biker
Top-k 1.5-entmax	fountain glass rides biker
Top-k sparsemax	fountain glass backpack wearing
Joint E and L	biker A fountain glass
Human highlights	a fountain
	(entailment)
	hypothesis: A clad in denim person works is working on a bridge outdoors.
	premise: A man wearing a denim jacket and jeans welds on a bridge above water while wearing his safety mask.
Erasure	bridge water safety welds
Top-k gradient	welds . water jeans
Top-k softmax	bridge water welds man
Top-k 1.5-entmax	bridge man welds on
Top-k sparsemax	bridge a jeans man
Joint E and L	bridge . safety water
Human highlights	denim jacket bridge
	(contradiction)
	hypothesis: The teenager is riding a horse in a steeplechase competition.
	premise: A teen is standing in a field and is in the upswing position after hitting a golf ball.
Erasure	golf ball field a
Top-k gradient	in golf hitting ball
Top-k softmax	golf ball field hitting
Top-k 1.5-entmax	ball field golf after
Top-k sparsemax	ball . after field
Joint E and L	A teen field ball
Human highlights	standing in a field hitting a golf ball

Table 7: Examples of extracted explanations for SNLI.