UNIREX: A Unified Learning Framework for Language Model Rationale Extraction

Aaron Chan [1] **Maziar Sanjabi** [2] **Lambert Mathias** [2] **Liang Tan** [2]
Shaoliang Nie [2] **Xiaochang Peng** [2] **Xiang Ren** [1] **Hamed Firooz** [2]

Abstract

An extractive rationale explains a language model's (LM's) prediction on a given task instance by highlighting the text inputs that most influenced the prediction. Ideally, rationale extraction should be *faithful* (reflective of LM's actual behavior) and *plausible* (convincing to humans), without compromising the LM's (*i.e.,* task model's) *task performance*. Although attribution algorithms and select-predict pipelines are commonly used in rationale extraction, they both rely on certain heuristics that hinder them from satisfying all three desiderata. In light of this, we propose UNIREX, a flexible learning framework which generalizes rationale extractor optimization as follows: (1) specify architecture for a learned rationale extractor; (2) select explainability objectives (*i.e.,* faithfulness and plausibility criteria); and (3) jointly train the task model and rationale extractor on the task using selected objectives. UNIREX enables replacing prior works' heuristic design choices with a generic learned rationale extractor in (1) and optimizing it for all three desiderata in (2)-(3). To facilitate comparison between methods w.r.t. multiple desiderata, we introduce the Normalized Relative Gain (NRG) metric. On five English text classification datasets, our best UNIREX configuration outperforms baselines by an average of 32.9% NRG. Plus, UNIREX rationale extractors' faithfulness can even generalize to unseen datasets and tasks.

1. Introduction

In recent years, neural language models (LMs) have yielded state-of-the-art performance on a wide range of natural language processing (NLP) tasks (Devlin et al., 2018; Liu et al., 2019). However, LMs' complex processes are notoriously opaque (Rudin, 2019), posing concerns about the societal implications of using LMs for high-stakes decision-making (Bender et al., 2021). Thus, explaining LMs' behavior is crucial for promoting trust, ethics, and safety in NLP systems (Doshi-Velez & Kim, 2017; Lipton, 2018). Given a LM's (*i.e.,* task model's) predicted label on a text classification instance, an *extractive rationale* is a type of explanation that highlights the tokens that most influenced the model to predict that label (Luo et al., 2021). To provide meaningful explanations, rationale extraction should be *faithful* (reflective of LM's actual behavior) (Ismail et al., 2021; Jain et al., 2020) and *plausible* (convincing to humans) (DeYoung et al., 2019), without compromising the LM's *task performance* (DeYoung et al., 2019; Jacovi & Goldberg, 2020) (Fig. 1).

Configuring the rationale extractor and its training process can greatly impact these desiderata, yet prior works have commonly adopted at least one of the following suboptimal heuristic design choices. First, many works rely in some way on *attribution algorithms* (AAs), which extract rationales via handcrafted functions (Sundararajan et al., 2017; Ismail et al., 2021; Situ et al., 2021). AAs may have built-in faithfulness-related properties but cannot be directly trained and tend to be compute-intensive (Bastings & Filippova, 2020). The most similar work to ours is SGT (Ismail et al., 2021), which regularizes a task model to produce faithful AA-based rationales. Still, AAs can be a bottleneck for plausibility, as producing human-like rationales is a complex objective requiring high capacity rationale extractors (Narang et al., 2020; DeYoung et al., 2019). Second, many works use a specialized *select-predict pipeline* (SPP), where a predictor module is trained to solve the task using only tokens chosen by a selector module (Jain et al., 2020; Yu et al., 2021; Paranjape et al., 2020). Instead of faithfulness optimization, SPPs heuristically aim for "faithfulness by construction" by treating the selected tokens as a rationale for the predictor's output (which depends only on those tokens). Still, SPPs typically have worse task performance than vanilla LMs since SPPs hide the full input from the predictor and are hard to train end-to-end (Jain et al., 2020; Bastings et al., 2019; Lei et al., 2016). Both AAs and SPPs

[1]University of Southern California [2]Meta AI. Correspondence to: Aaron Chan <chanaaro@usc.edu>.

Proceedings of the 39^{th} International Conference on Machine Learning, Baltimore, Maryland, USA, PMLR 162, 2022.

utilize heuristics that fundamentally limit the rationale extractor from achieving all three desiderata.

To tackle this challenge, we propose the **UNI**fied Learning Framework for **R**ationale **EX**traction (**UNIREX**), which generalizes rationale extractor optimization as follows: (1) specify architecture for a learned rationale extractor; (2) select explainability objectives (*i.e.,* faithfulness and plausibility criteria); and (3) jointly train the task model and rationale extractor on the task using selected objectives (Sec. 3). UNIREX enables replacing prior works' heuristic design choices in (1) with a generic learned rationale extractor and optimizing it for all three desiderata in (2)-(3).

UNIREX provides great flexibility in performing (1)-(3). For (1), any model architecture is applicable, but we study Transformer LM based rationale extractors in this work (Zaheer et al., 2020; DeYoung et al., 2019). We focus on two architectures: (A) Dual LM, where task model and rationale extractor are separate; and (B) Shared LM, where task model and rationale extractor share parameters. For (2), any faithfulness and plausibility criteria can be used. Following DeYoung et al. (2019), we focus on comprehensiveness and sufficiency as faithfulness criteria, while using similarity to gold rationales as plausibility criteria. For (3), tradeoffs between the three desiderata can be easily managed during rationale extractor optimization by setting arbitrary loss weights for the faithfulness and plausibility objectives. Furthermore, although computing the faithfulness criteria involves discrete (non-differentiable) token selection, using the Shared LM architecture can approximate end-to-end training and enable both task model and rationale extractor to be optimized w.r.t. all three desiderata (Sec. 3.4).

To evaluate all three desiderata in aggregate, we introduce the Normalized Relative Gain (NRG) metric. On five English text classification datasets – SST, Movies, CoS-E, MultiRC, and e-SNLI (Carton et al., 2020; DeYoung et al., 2019) – our best UNIREX configuration outperforms the strongest baselines by an average of 32.9% NRG (Sec. 4.3), showing that UNIREX can optimize rationale extractors for all three desiderata. In addition, we verify our UNIREX design choices via extensive ablation studies (Sec. A.3). Moreover, UNIREX-trained extractors have considerable generalization power, yielding high plausiblity with minimal gold rationale supervision (Sec. 4.4) and high faithfulness on unseen datasets/tasks (Sec. 4.5). Finally, our user study shows that humans judge UNIREX rationales as more plausible than rationales extracted via other methods (Sec. A.5).

2. Problem Formulation

We formalize rationale extraction and discuss how extracted rationales are evaluated, in the context of text classification.

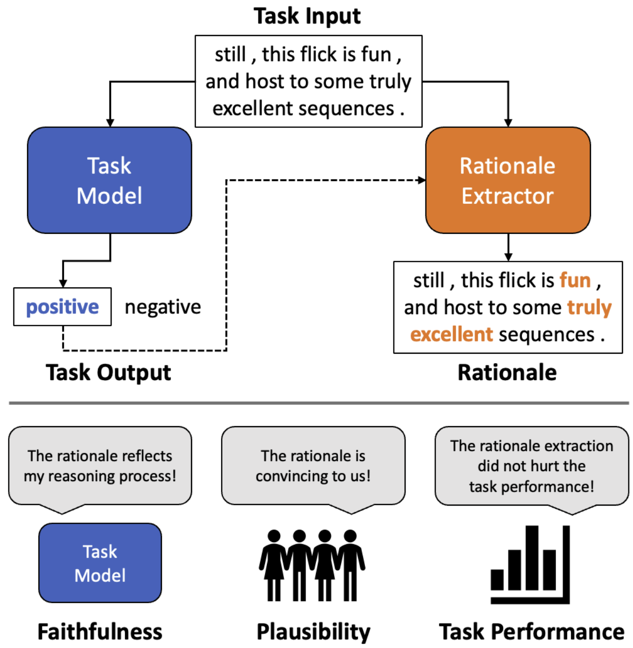

Figure 1: **Desiderata of Rationale Extraction.** Ideally, rationale extraction should be *faithful* and *plausible*, without compromising the task model's *task performance*. Unlike prior works, UNIREX enables optimizing the rationale extractor for all three desiderata.

2.1. Rationale Extraction

Here, we consider $\mathcal{F}_{\text{task}} = f_{\text{task}}(f_{\text{enc}}(\cdot))$ as a task model for M-class text classification (Sec. A.2), where f_{enc} is the text encoder while f_{task} is the task output head. In modern NLP systems, $\mathcal{F}_{\text{task}}$ usually has a BERT-style architecture (Devlin et al., 2018), in which f_{enc} is a Transformer network (Vaswani et al., 2017) while f_{task} is a linear layer with softmax classifier. Let $\mathbf{x}_i = [x_i^t]_{t=1}^n$ be the n-token input sequence (*e.g.,* a sentence) for task instance i, and $\mathcal{F}_{\text{task}}(\mathbf{x}_i) \in \mathbb{R}^M$ be the logit vector for the output of the task model. We use $y_i = \arg\max_j \mathcal{F}_{\text{task}}(\mathbf{x}_i)_j$ to denote the class predicted by $\mathcal{F}_{\text{task}}$. Given $\mathcal{F}_{\text{task}}$, $\mathbf{x}_i$, and y_i, the goal of rationale extraction is to output vector $\mathbf{s}_i = [s_i^t]_{t=1}^n \in \mathbb{R}^n$, such that each $s_i^t \in \mathbb{R}$ is an *importance score* indicating how strongly token x_i^t influenced $\mathcal{F}_{\text{task}}$ to predict class y_i.

Let $\mathcal{F}_{\text{ext}}$ denote a rationale extractor, such that $\mathbf{s}_i = \mathcal{F}_{\text{ext}}(\mathcal{F}_{\text{task}}, \mathbf{x}_i, y_i)$. $\mathcal{F}_{\text{ext}}$ can be a learned or heuristic function. In practice, the final rationale is typically obtained by binarizing $\mathbf{s}_i$ as $\mathbf{r}_i \in \{0, 1\}^n$, via the top-$k$% strategy: $r_i^t = 1$ if s_i^t is one of the top-k% scores in $\mathbf{s}_i$; otherwise, $r_i^t = 0$ (DeYoung et al., 2019; Jain et al., 2020; Pruthi et al., 2020; Chan et al., 2021). While other binarization strategies can be used (*e.g.,* score threshold, highest-scoring contiguous k-token span), we focus on top-k% in this study, since this strategy is most prevalent. For top-k%, let $\mathbf{r}_i^{(k)}$ denote the "important" (*i.e.,* ones) tokens in $\mathbf{r}_i$, when using $0 \leq k \leq 100$.

2.2. Three Desiderata of Rationale Extraction

To provide meaningful explanations, rationale extraction via F_{ext} should be *faithful* and *plausible*, without significantly hurting F_{task}'s *task performance* (DeYoung et al., 2019).

Faithfulness Faithfulness means how accurately a rationale reflects F_{task}'s true reasoning process for predicting y_i (Jacovi & Goldberg, 2020). Hence, faithfulness metrics aim to measure the extent to which the $\mathbf{r}_i^{(k)}$ tokens influence $p_{y_i}(\mathbf{x}_i)$, which denotes F_{task}'s confidence probability for y_i when using $\mathbf{x}_i$ as input (DeYoung et al., 2019; Shrikumar et al., 2017; Hooker et al., 2018; Pruthi et al., 2020). Recently, comprehensiveness and sufficiency have emerged as popular faithfulness metrics in the explainability literature (DeYoung et al., 2019). ***Comprehensiveness*** (comp) measures the change in p_{y_i} when $\mathbf{r}_i^{(k)}$ is *removed* from the input: comp $= p_{y_i}(\mathbf{x}_i) - p_{y_i}(\mathbf{x}_i \backslash \mathbf{r}_i^{(k)})$. That is, if the $\mathbf{r}_i^{(k)}$ tokens are truly influential, then removing them from the input should decrease F_{task}'s predicted probability for y_i. Thus, higher comp indicates higher faithfulness. ***Sufficiency*** (suff) measures the change in p_{y_i} when only $\mathbf{r}_i^{(k)}$ is *kept* in the input: suff $= p_{y_i}(\mathbf{x}_i) - p_{y_i}(\mathbf{r}_i^{(k)})$. That is, if the $\mathbf{r}_i^{(k)}$ tokens are truly influential, only keeping them in the input should not decrease F_{task}'s predicted probability for y_i. Thus, lower suff indicates higher faithfulness.

Plausibility Plausibility is defined as how convincingly a rationale explains a given model's prediction, as judged by humans (Jacovi & Goldberg, 2020). This can be measured either by automatically computing the similarity between F_{ext}'s rationales (either $\mathbf{s}_i$ or $\mathbf{r}_i$) and human-annotated gold rationales (DeYoung et al., 2019), or by asking human annotators to rate whether F_{ext}'s rationales make sense for predicting y_i (Strout et al., 2019; Doshi-Velez & Kim, 2017). Typically, a gold rationale is a binary vector $\mathbf{r}_i \in \{0, 1\}^n$, where ones and zeros indicate important and unimportant tokens, respectively (Lei et al., 2016; DeYoung et al., 2019).

Task Performance Task performance, in the context of rationale extraction, concerns how much F_{task}'s task performance (on the test set) drops when F_{task} is trained with explainability objectives (*i.e.,* faithfulness, plausibility) for F_{ext}. As long as F_{task} is trained with non-task losses, F_{task}'s task performance can be affected. Note that this means post hoc (*i.e.,* introduced after F_{task} training is over) rationale extraction will not affect F_{task}'s task performance. In general, the main goal of F_{task} is high task performance, so we should ideally improve F_{ext} w.r.t. the other desiderata without hurting F_{task}'s task performance. To measure task performance, we use standard dataset-specific performance metrics (*e.g.,* accuracy, F1).

3. UNIREX

We present the UNIREX learning framework, which enables jointly optimizing the task model and rationale extractor, w.r.t. faithfulness, plausibility, and task performance.

3.1. Framework Overview

Given task model F_{task}, UNIREX generalizes rationale extractor optimization as follows: (1) choose architecture for a learned rationale extractor F_{ext}; (2) select explainability objectives (*i.e.,* faithfulness loss $\mathcal{L}_{\text{faith}}$ and plausibility loss $\mathcal{L}_{\text{plaus}}$); and (3) jointly train F_{task} and F_{ext} using $\mathcal{L}_{\text{task}}$ (task loss), $\mathcal{L}_{\text{faith}}$, and $\mathcal{L}_{\text{plaus}}$. As shown in Fig. 2, UNIREX training consists of two backpropagation paths. The first path is used to update F_{task} w.r.t. $\mathcal{L}_{\text{task}}$ and $\mathcal{L}_{\text{faith}}$. Whereas $\mathcal{L}_{\text{task}}$ is computed w.r.t. the task target y_i, $\mathcal{L}_{\text{faith}}$ is computed only using the task input $\mathbf{x}_i$ and the top-$k\%$ important tokens $\mathbf{r}_i^{(k)}$ (obtained via F_{ext}), based on some combination of comp and suff (Sec. 2.2). The second path is used to update F_{ext} w.r.t. $\mathcal{L}_{\text{plaus}}$, which encourages importance scores $\mathbf{s}_i$ to approximate gold rationale $\mathbf{r}_i$.

Thus, UNIREX frames rationale extraction as the following optimization problem:

$$
\begin{aligned}
\min_{F_{\text{task}},\, F_{\text{ext}}} \; & \mathcal{L}_{\text{task}}(\mathbf{x}_i, y_i; F_{\text{task}}) \\
& + \lambda_f \mathcal{L}_{\text{faith}}(\mathbf{x}_i, \mathbf{r}_i^{(k)}; F_{\text{task}}) \\
& + \lambda_p \mathcal{L}_{\text{plaus}}(\mathbf{x}_i, \mathbf{r}_i; F_{\text{ext}}),
\end{aligned} \tag{1}
$$

where λ_f and λ_p are loss weights. If F_{task} and F_{ext} share parameters, then the shared parameters will be optimized w.r.t. all losses. During inference, for task input $\mathbf{x}_i$, we first use F_{task} to predict y_i, then use F_{ext} to output a rationale $\mathbf{r}_i$ for F_{task}'s prediction y_i. Below, we discuss options for UNIREX's rationale extractor and explainability objectives.

3.2. Rationale Extractor

In UNIREX, F_{ext} is a learned function by default. Here, we first introduce heuristic F_{ext} (*i.e.,* AA), then discuss why a learned F_{ext} should typically be preferred (Sec. 2.1). For each F_{ext} type, we present several possible design choices and the pros/cons of the given type.

3.2.1. Heuristic Rationale Extractors

Heuristic F_{ext} refers to AAs, which can be any handcrafted function that calculates an importance score s_i^t for each input token x_i^t (Bastings & Filippova, 2020). AAs are typically gradient-based (Sundararajan et al., 2017; Denil et al., 2014; Lundberg & Lee, 2017; Li et al., 2015) or perturbation-based (Li et al., 2016; Poerner et al., 2018; Kádár et al., 2017) methods. Recall that $p_{y_i}(\mathbf{x}_i)$ denotes F_{task}'s predicted probability for class y_i (Sec. 2.2). Gradient-based methods compute s_i^t via the gradient of $p_{y_i}(\mathbf{x}_i)$ w.r.t. x_i^t.

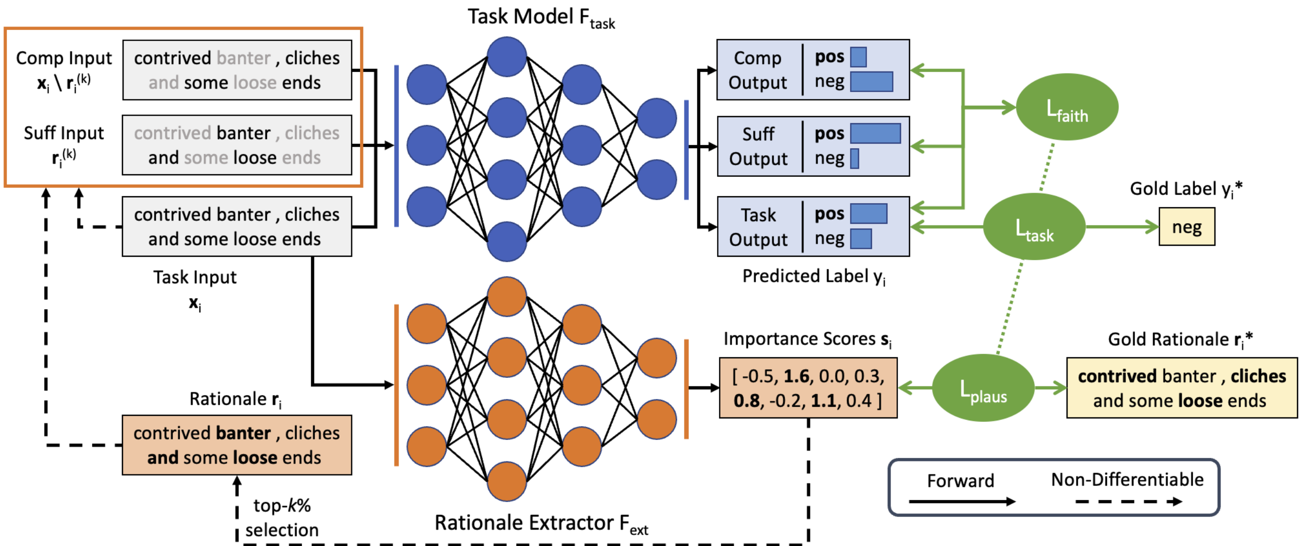

Figure 2: **UNIREX Framework.** UNIREX enables jointly optimizing the task model (F_{task}) and rationale extractor (F_{ext}), w.r.t. faithfulness (L_{faith}), plausibility (L_{plaus}), and task performance (L_{task}). In this example, we consider the sentiment analysis task. For **task performance**, F_{task} is trained via gold label y_i^* to predict the sentiment – either positive (pos) or negative (neg) – of sentence $\mathbf{x}_i$. Here, F_{task}'s predicted label for $\mathbf{x}_i$ is y_i = pos. For **plausibility**, F_{ext} is trained via gold rationale $\mathbf{r}_i^*$ to output human-aligned token importance scores $\mathbf{s}_i$ for $\mathbf{x}_i$ (Sec. 3.3.2). For **faithfulness**, $\mathbf{s}_i$ is binarized as rationale $\mathbf{r}_i$ via top-$k\%$ selection, then used to construct the comp ($\mathbf{x}_i n \mathbf{r}_i^{(k)}$) and suff ($\mathbf{r}_i^{(k)}$) inputs for L_{task}. With L_{task}'s predicted probabilities for y_i, given $\mathbf{x}_i$, $\mathbf{x}_i n \mathbf{r}_i^{(k)}$, and $\mathbf{r}_i^{(k)}$, respectively, the comp and suff losses are computed. The comp and suff losses align L_{task}'s output with $\mathbf{r}_i$, such that $\mathbf{r}_i$ becomes a faithful explanation of L_{task}'s behavior. (Sec. 3.3.1). Note that some parts of UNIREX are non-differentiable. Still, by having L_{task} and L_{ext} share a text encoder, we can approximate end-to-end training of both models, jointly w.r.t. all three desiderata (Sec. (3.4).

These methods require one or more F_{task} backward passes. Perturbation-based methods measure s_i^t as $p_{y_i}(\mathbf{x}_i)$'s change when perturbing (*e.g.,* removing) x_i^t. These methods require multiple F_{task} forward passes – typically, one forward pass per token in $\mathbf{x}_i$.

AAs can be used out of the box without training and are designed to satisfy certain faithfulness-related axiomatic properties (Sundararajan et al., 2017; Lundberg & Lee, 2017). However, AAs' lack of learnable parameters means they cannot be optimized for faithfulness/plausibility. Thus, if F_{task} is trained for explainability using AA-based rationales, then only F_{task} is optimized. Also, faithful AAs tend to be compute-intensive, requiring many F_{task} backward/forward passes per instance (Sundararajan et al., 2017; Lundberg & Lee, 2017; Li et al., 2016).

3.2.2. Learned Rationale Extractors

Learned F_{ext} can be any learned model that transforms x_i^t into s_i^t. Given their success in NLP explainability (DeYoung et al., 2019), we focus on pre-trained Transformer LMs and highlight two key architectures: Dual LM (DLM) and Shared LM (SLM) (Fig. 6). For DLM, F_{task} and F_{ext} are two separate Transformer LMs with the same encoder architecture. Formally, we define the DLM extractor as $F_{\text{ext}} = f_{\text{ext}}(f^{\emptyset}_{\text{enc}}(\,))$, where $f^{\emptyset}_{\text{enc}}$ and f_{ext} are F_{ext}'s encoder and output head, respectively. DLM provides more capacity for F_{ext}, which can help F_{ext} output plausible rationales. For SLM, F_{task} and F_{ext} are two Transformer LMs sharing encoder f_{enc}, while F_{ext} has its own output head f_{ext}. Formally, the SLM extractor is defined as $F_{\text{ext}} = f_{\text{ext}}(f_{\text{enc}}(\,))$. SLM leverages multitask learning between F_{task} and F_{ext}, which can improve faithfulness since F_{ext} has greater access to information about F_{task}'s reasoning process. By default, F_{ext} takes $\mathbf{x}_i$ as input and uses a linear layer for f_{ext}, although these settings can be changed if desired.

Unlike heuristic F_{ext}, learned F_{ext} can be optimized for faithfulness/plausibility and only require one F_{task} forward pass during inference (*e.g.,* perturbation-based AAs require n forward passes per n-token instance). However, they cannot be used out of the box without explainability training and do not have built-in axiomatic properties – *e.g.,* sensitivity and implementation invariance in the IG (Sundararajan et al., 2017) AA – which are designed to promote faithfulness.

Overall, learned F_{ext} is preferred if: (A) the goal is to optimize for both faithfulness and plausibility, and (B) gold rationales – even a small amount – are available for plausibility optimization. (B) is true because gold rationale annotated instances can be provided in every batch via oversampling (Sec A.4), which works surprisingly well in low-resource settings (Sec. 4.4). Otherwise, UNIREX allows for the learned F_{ext} to be replaced with a heuristic F_{ext}.

3.3. Explainability Objectives

After selecting F_{ext}, we specify the explainability objectives, which can be any combination of faithfulness and plausibility criteria. In prior approaches (*e.g.,* AA, SPPs), the rationale extractor is not optimized for both faithfulness and plausibility, but UNIREX makes this possible. For any choice of learned F_{ext}, UNIREX lets us easily "plug and play" different criteria and loss weights, based on our needs and domain knowledge, to find those that best balance the rationale extraction desiderata.

3.3.1. Faithfulness

Faithfulness refers to how accurately a rationale (output by F_{ext}) reflects F_{task}'s decision process for a given instance. Evaluating rationale faithfulness is still an open problem with numerous applicable metrics, and UNIREX is not tailored for any specific metric. However, given the prevalence of comp and suff (Sec. 2.1) in the explainability literature (DeYoung et al., 2019; Ismail et al., 2021), we focus on comp and suff related objectives.

Recall that comp measures the importance of tokens in $\mathbf{r}_i^{(k)}$ as how $p_{y_i}(\mathbf{x}_i)$ changes when those tokens are removed from $\mathbf{x}_i$. Intuitively, we want $p_{y_i}(\mathbf{x}_i)$ to be higher than $p_{y_i}(\mathbf{x}_i \setminus \mathbf{r}_i^{(k)})$, so higher comp is better. Since comp is defined for a single class' probability rather than the label distribution, we can define the comp loss L_{comp} via cross-entropy loss L_{CE} (which is computed w.r.t. the target class), as in the following *difference criterion* instantiation of L_{comp}:

$$L_{\text{comp-diff}} = L_{\text{CE}}(F_{\text{task}}(\mathbf{x}_i), y_i) - L_{\text{CE}}(F_{\text{task}}(\mathbf{x}_i \setminus \mathbf{r}_i^{(k)}), y_i) \tag{2}$$

$$L_{\text{CE}}(F_{\text{task}}(\mathbf{x}_i), y_i) = -y_i \log(F_{\text{task}}(\mathbf{x}_i)) \tag{3}$$

For training stability, we compute comp loss for target class y_i here instead of F_{task}'s predicted class $\hat{y}_i$, since $\hat{y}_i$ is a moving target during training. Using $L_{\text{comp-diff}}$, it is possible for $L_{\text{CE}}(F_{\text{task}}(\mathbf{x}_i \setminus \mathbf{r}_i^{(k)}), y_i))$ to become much larger than $L_{\text{CE}}(F_{\text{task}}(\mathbf{x}_i), y_i)$, leading to arbitrarily negative losses. To prevent this, we can use margin m_c to impose a lower bound on $L_{\text{comp-diff}}$, yielding the following *margin criterion*:

$$L_{\text{comp-margin}} = \max(-m_c, L_{\text{comp-diff}}) + m_c \tag{4}$$

Recall that suff measures the importance of tokens in $\mathbf{r}_i^{(k)}$ as how $p_{y_i}(\mathbf{x}_i)$ changes when they are the only tokens kept in $\mathbf{x}_i$. Based on suff's definition, we want $p_{y_i}(\mathbf{r}_i^{(k)})$ to be higher than $p_{y_i}(\mathbf{x}_i)$, so lower suff is better. For suff loss L_{suff}, we define the difference and margin criteria analogously to L_{comp}, using margin m_s but the opposite sign for $L_{\text{suff-diff}}$ (since lower suff is better):

$$L_{\text{suff-diff}} = L_{\text{CE}}(F_{\text{task}}(\mathbf{r}_i^{(k)}), y_i) - L_{\text{CE}}(F_{\text{task}}(\mathbf{x}_i), y_i) \tag{5}$$

$$L_{\text{suff-margin}} = \max(-m_s, L_{\text{suff-diff}}) + m_s \tag{6}$$

In our experiments, we find that the margin-based comp and suff criteria are effective (Sec. A.3), though others (*e.g.,* KL divergence, MAE) can be used too (Sec. A.6.1). Note that $\mathbf{r}_i^{(k)}$ is computed via top-k% thresholding (Sec. 2.1), so we also need to specify a set K of threshold values. We separately compute the comp and suff losses for each $k \in K$, then obtain the final comp and suff losses by averaging over all k values via area-over-precision-curve (AOPC) (DeYoung et al., 2019). To reflect this, we denote the comp and suff losses as $L_{\text{comp},K}$ and $L_{\text{suff},K}$, respectively. Let $\lambda_f L_{\text{faith}} = \lambda_c L_{\text{comp},K} + \lambda_s L_{\text{suff},K}$, where λ_c and λ_s are loss weights. In this case, we can abstractly consider λ_f as an aggregate loss weight for the faithfulness objectives.

3.3.2. Plausibility

Plausibility is defined as how convincing a rationale (output by F_{ext}) is to humans as an explanation for F_{task}'s prediction on a given instance (Jacovi & Goldberg, 2020). Since F_{task}'s predictions may change throughout its training, optimizing for plausibility should ideally involve continual human-in-the-loop feedback. However, obtaining such human-in-the-loop feedback is prohibitive, so many works consider human-annotated gold rationales as a cheaper form of plausibility supervision (DeYoung et al., 2019; Narang et al., 2020; Jain et al., 2020). Even so, gold rationales $\mathbf{r}_i^*$ are generally only annotated w.r.t. the gold task label y_i (as opposed to F_{task}'s predicted label $\hat{y}_i$, which cannot be known *a priori*). Consequently, if $y_i \neq \hat{y}_i$, then gold rationale supervision may be noisy.

Still, this is not a significant issue if F_{task} is jointly trained with F_{ext}. In UNIREX, F_{task} and F_{ext} are jointly trained to predict y_i (via L_{task}) and $\mathbf{r}_i^*$ (via L_{plaus}), respectively, while F_{task} is also regularized (via L_{faith}) such that its output $\hat{y}_i$ aligns with F_{ext}'s output $\mathbf{r}_i$. In other words, y_i may be an acceptable approximation of $\hat{y}_i$ when training F_{ext} to predict $\mathbf{r}_i^*$ (which is based on y_i) because: (A) F_{task} is jointly trained such that its output $\hat{y}_i$ approximates y_i, and (B) F_{ext} is also trained such that its output $\mathbf{r}_i$ aligns with $\hat{y}_i$. As a result, if gold rationale supervision is available, then we can optimize for plausibility via UNIREX. Specifically, given gold rationale $\mathbf{r}_i^*$ for input $\mathbf{x}_i$, plausibility optimization entails training F_{ext} to predict binary importance label $\mathbf{r}_i^{*,t}$ for each token x_i^t. This is essentially binary token classification, so one natural choice for L_{plaus} is the token-level binary cross-entropy (BCE) criterion:

$$L_{\text{plaus-BCE}} = -\sum_t \mathbf{r}_i^{*,t} \log(F_{\text{ext}}(x_i^t)) \tag{7}$$

Besides BCE loss, we can also consider other criteria like sequence-level KL divergence and linear loss. See Sec. A.6.2 for discussion of these and other plausibility criteria.

3.4. Training and Inference

After setting F_{ext}, L_{faith}, and L_{plaus}, we can move on to training F_{task} and F_{ext}. Since top-k% rationale binarization (Sec. 3.3) is not differentiable, by default, we cannot backpropagate L_{faith} through all of F_{ext}'s parameters. Thus, F_{task} is trained via L_{task} and L_{faith}, while F_{ext} is only trained via L_{plaus}. This means F_{ext}'s rationales $\mathbf{r}_i$ are indirectly optimized for faithfulness by regularizing F_{task} such that its behavior aligns with $\mathbf{r}_i$. The exception is if we are using the SLM variant, where encoder f_{enc} is shared by F_{task} and F_{ext}. In this case, f_{enc} is optimized w.r.t. all losses, task head f_{task} is optimized w.r.t. L_{task} and L_{faith}, and extractor head f_{ext} is optimized w.r.t. L_{plaus}. SLM is a simple way to approximate end-to-end training of F_{task} and F_{ext}. In contrast, past SPPs have used more complex methods like reinforcement learning (Lei et al., 2016) and the reparameterization trick (Bastings et al., 2019), whose training instability has been shown hurt task performance (Jain et al., 2020).

Now, we summarize the full learning objective. Given that cross-entropy loss $L_{\text{task}} = L_{\text{CE}}(F_{\text{task}}(\mathbf{x}_i), y_i)$ is used to train F_{task} to predict y_i, the full learning objective is:

$$\begin{aligned} L &= L_{\text{task}} + \alpha_f L_{\text{faith}} + \alpha_p L_{\text{plaus}} \\ &= L_{\text{task}} + \alpha_c L_{\text{comp},K} + \alpha_s L_{\text{suff},K} + \alpha_p L_{\text{plaus}}. \end{aligned} \tag{8}$$

During inference, we use F_{task} to predict y_i, then use F_{ext} to output $\mathbf{r}_i$ for F_{task}'s predicted label y_i.

4. Experiments

We present empirical results showing UNIREX's effectiveness in trading off faithfulness, plausibility, and task performance during rationale extractor optimization. First, our main experiments compare rationale extraction methods w.r.t. all three desiderata (Sec. 4.3). Second, we perform various ablation studies to verify our design choices for UNIREX (Sec. A.3). Third, we present experiments showing UNIREX's strong data efficiency, w.r.t. limited gold rationale supervision (Sec. 4.4) and zero-shot faithfulness transfer (Sec. 4.5). Fourth, to account for the limitations of gold-rationale-based plausibility evaluation, we conduct a user study to further demonstrate the improved plausibility of UNIREX-extracted rationales (Sec. A.5).

4.1. Evaluation Protocol

4.1.1. Datasets

We primarily experiment with the SST (Socher et al., 2013; Carton et al., 2020), Movies (Zaidan & Eisner, 2008), CoS-E (Rajani et al., 2019), MultiRC (Khashabi et al., 2018), and e-SNLI (Camburu et al., 2018) datasets, all of which have gold rationale annotations. The latter four datasets were taken from the ERASER benchmark (DeYoung et al., 2019). For the zero-shot faithfulness transfer experiments (Sec. 4.5), we consider five additional datasets, which are described further in Sec. 4.5.

4.1.2. Metrics

To measure faithfulness, plausibility, and task performance, we use the metrics from the ERASER benchmark (DeYoung et al., 2019). For faithfulness, we use comp and suff, for $k = [1, 5, 10, 20, 50]$ (DeYoung et al., 2019). For plausibility, we use area under precision-recall curve (AUPRC) and token F1 (TF1) to measure similarity to gold rationales (DeYoung et al., 2019; Narang et al., 2020). For task performance, we follow the dataset-specific metrics used in ERASER: accuracy for SST and CoS-E; macro F1 for Movies, MultiRC, and e-SNLI (DeYoung et al., 2019). That is, we only use one task performance metric per dataset.

Normalized Relative Gain (NRG) After computing these raw metrics for faithfulness, plausibility, and task performance, we would like to compare different rationale extraction methods w.r.t. all three desiderata. However, aggregating the raw metrics across the three desiderata may not be straightforward. In light of this, we introduce the Normalized Relative Gain (NRG) metric, which is based on the Average Relative Gain (ARG) metric (Ye et al., 2021). For each raw metric, NRG transforms all raw scores to normalized scores in $[0, 1]$ (higher is better). After all raw metrics are in the same $[0, 1]$ space, we can simply aggregate them via averaging. We formally describe this process below.

For each raw metric (*e.g.,* comp, suff, AUPRC, accuracy), we are given a set of raw scores $Z = \{z_1, z_2, ...\}$. Each raw score $z_i \in Z$ corresponds to a different rationale extraction method i. NRG(z_i) captures z_i's relative gain over the worst score in Z, normalized w.r.t. score range $\max(Z) - \min(Z)$. The definition of "worst score" depends on whether higher or lower raw scores are better for the given metric. If higher raw scores are better (*e.g.,* comp, AUPRC, accuracy), then the worst score would be $\min(Z)$, which yields: NRG(z_i) $= \frac{z_i - \min(Z)}{\max(Z) - \min(Z)}$. If lower values are better (*e.g.,* sufficiency), then the worst score would be $\max(Z)$, which yields: NRG(z_i) $= \frac{\max(Z) - z_i}{\max(Z) - \min(Z)}$.

After computing the individual NRG for each raw metric, we obtain the desiderata NRG scores by averaging the individual NRG scores within each desideratum. Let FNRG, PNRG, and TNRG be the desiderata NRG scores for faithfulness, plausibility, and task performance, respectively. FNRG is the average of the individual NRG scores for comp and suff; PNRG is the average of the individual NRG scores for AUPRC and TF1; and TNRG is just the individual NRG for the task performance metric (since there is only one task performance metric per dataset). Finally, to summarize all of the raw metrics as a single score, we compute the composite NRG (CNRG) by averaging the three desiderata NRG scores: CNRG = $\frac{\text{FNRG}+\text{PNRG}+\text{TNRG}}{3}$. By default, we compute CNRG as an unweighted average of the three desiderata NRG scores, under the assumption that all three desiderata are equally important. On the other hand, for situations where certain desiderata are more important than others, we can also compute CNRG as a weighted average.

Generally, the computation of NRG should involve globally aggregating the raw metrics across all available methods, which is done in the main results (Sec. 4.3). However, for a number of more focused experiments (Sec. A.3 and 4.5), only a subset of the available methods are considered. Thus, for these experiments, we report the raw metrics instead of NRG (Tables 2 and 1).

4.1.3. Results Reporting

For all results, we report the average over three seeds and five k faithfulness thresholds (*i.e.,* $k = [1, 5, 10, 20, 50]$), a total of 15 settings. We denote each UNIREX configuration with a parenthetical "([*rationale extractor*]-[*explainability objectives*])". For the rationale extractor, AA, DLM, and SLM denote attribution algorithm, Dual LM, and Shared LM, respectively. For explainability objectives, F, P, and FP denote faithfulness, plausibility, and faithfulness+plausibility, respectively. For example, DLM-FP means Dual LM with faithfulness+plausibility objectives.

4.2. Baselines

We consider a wide range of representative rationale extraction baselines, spanning three key categories. Note that some methods do not assume access to gold rationales, which prevents such methods from optimizing for plausibility. This means that not all of the methods are directly comparable. Therefore, when comparing methods, we generally group them by whether they use optimize for plausibility and only compare methods within the same group.

The first category is **vanilla attribution algorithm (AA)**, which does not involve training F_{ext} and is applied post hoc (*i.e.,* they do not impact F_{task}'s training). Included baselines from this category are: Gradient (Grad) (Simonyan et al., 2013), Input*Gradient (Input*Grad) (Denil et al., 2014), DeepLIFT (Lundberg & Lee, 2017), and Integrated Gradients (IG) (Sundararajan et al., 2017). These four baselines are among the most popular AAs in the explainability literature (Luo et al., 2021; Pruthi et al., 2020). In the results, we denote these baselines as "AA ([*AA name*])", e.g., AA (IG).

The second category is **AA-based training**, which uses AAs in some way to train a learned F_{ext}. One baseline in this category is L2E (Situ et al., 2021), which distills knowledge from an AA to an LM-based F_{ext}. Specifically, after training F_{task}, then using an AA to extract rationales for F_{task}, L2E entails training F_{ext} to output rationales that are similar to the AA's rationales. Another baseline is SGT (Ismail et al., 2021), which uses a suff-based criterion to regularize F_{task}, such that the AA yields faithful rationales for F_{task}. We also consider a variant called SGT+P, which augments SGT with plausibility optimization via gold rationales. For all baselines in this category, we use IG as the AA.

The third category is **select-predict pipeline (SPP)**, where F_{task} (predictor) only takes input tokens chosen via F_{ext}'s (selector) rationale output. One baseline in this category is FRESH (Jain et al., 2020), which trains F_{task} and F_{ext} separately. For FRESH, we use a stronger variant (compared to those in the FRESH paper) where IG rationales are directly provided to the predictor, rather than output by a trained F_{ext}. Another baseline is A2R (Yu et al., 2021), a recently proposed SPP which aims to improve F_{task}'s task performance by regularizing F_{task} with an attention-based predictor that uses the full input. Also, we introduce FRESH+P and A2R+P, which respectively augment FRESH and A2R with plausibility optimization.

4.3. Main Results

Figs. 3-7 display the main results, in terms of NRG. For conciseness, we omit AA (Grad), AA (Input*Grad), and AA (DeepLIFT) results from these NRG figures, since AA (IG) is representative of vanilla AA methods. Please refer to Sec. A.11 for all raw and NRG empirical results.

In Figs. 3-4, we use CNRG to compare rationale extraction methods for each dataset. Here, the Dataset Mean group reports the mean CNRG across all datasets. First, Fig. 3 compares methods that *do not* optimize for plausibility (since they do not have access to gold rationales). Overall, we find that UNIREX (AA-F) achieves the best CNRG on Dataset Mean (and on all datasets except Movies), showing the effectiveness of UNIREX's faithfulness optimization. On Dataset Mean, UNIREX (AA-F) beats the strongest baseline (*i.e.,* SGT) by 9.2%. Second, Fig. 4 compares methods that *do* optimize for plausibility. Overall, we find that UNIREX (SLM-FP) and UNIREX (DLM-FP) achieve the best CNRG – both beating the strongest baseline (*i.e.,* A2R+P) by over 30% on Dataset Mean – demonstrating UNIREX's ability to jointly optimize F_{task} and F_{ext} for all three desiderata. Meanwhile, UNIREX (DLM-P) performs slightly worse but still significantly better than all baselines, showing the effectiveness of UNIREX's plausibility optimization.

Fig. 7 compares rationale extraction methods w.r.t. the desiderata NRG, averaged over all datasets. First, Fig. 7 (**left**) compares rationale extraction methods *without* plausibility optimization, so PNRG is low for all methods here. Here, we see that UNIREX (AA-F)'s FNRG is highest, while its TNRG is close to highest. As shown in Fig. 3, UNIREX (AA-F) achieves the best composite NRG

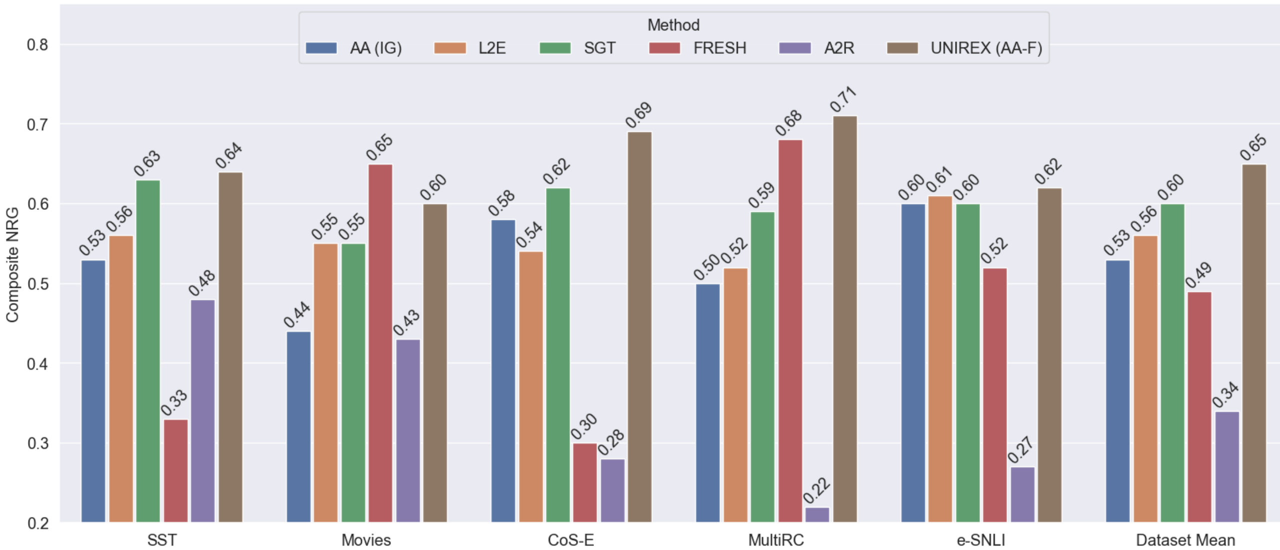

Figure 3: **Composite NRG Comparison (w/o Plausibility Optimization).** The composite NRG (CNRG) is the mean of the three desiderata NRG scores. For each dataset, we use CNRG to compare rationale extraction methods that *do not* optimize for plausibility. Overall, UNIREX (AA-F) achieves the best CNRG on Dataset Mean (and on all datasets except Movies), showing the effectiveness of UNIREX's faithfulness optimization. On Dataset Mean, UNIREX (AA-F) beats the strongest baseline (*i.e.,* SGT) by 9.2%.

(CNRG) because UNIREX training enables effective balancing of faithfulness and task performance. On the other hand, baselines with high FNRG (*i.e.,* FRESH, A2R) have low TNRG, while baselines with high TNRG (*i.e.,* AA (IG), L2E, SGT) have low FNRG. Second, Fig. 7 (**right**) compares rationale extraction methods *with* plausibility optimization. Here, we see that UNIREX (DLM-FP) and UNIREX (SLM-FP) have moderate FNRG, but the highest (or near-highest) PNRG and TNRG. Meanwhile, UNIREX (DLM-P) achieves the highest PNRG and TNRG, but the worst FNRG, since UNIREX (DLM-P) does not optimize for faithfulness. As shown in Fig. 4, UNIREX (DLM-FP) and UNIREX (SLM-FP) achieve the best CNRG because UNIREX training enables effective balancing of faithfulness and task performance. Meanwhile, baselines with high FNRG (*i.e.,* FRESH+P, A2R+P) have low TNRG, while baselines with high TNRG (*i.e.,* SGT+P) have low PNRG.

4.4. Gold Rationale Efficiency

UNIREX supports arbitrary amounts of gold rationale supervision, allowing plausibility optimization even in low-resource settings. In Fig. 5, we compare plausibility (w.r.t. AUPRC) for $= [0.5; 1; 5; 10; 20; 100]$ (*i.e.,* % of train instances with gold rationales). We compare AA (IG) and four UNIREX variants (AA-F, AA-FP, DLM-FP, SLM-FP), with standard deviation shown by the error bands. First, AA (IG) and UNIREX (AA-F) do not optimize for plausibility via gold rationales, so their low AUPRC scores are constant for all . Second, UNIREX (AA-FP)'s AUPRC varies directly with at a modest rate, but is still always lower than AA (IG)'s and UNIREX (AA-F)'s AUPRC. UNIREX (AA-FP)'s plausibility optimization is not effective, since plausibility optimization (*i.e.,* learning to generate human-like rationales) typically requires high learning capacity, yet AAs do not have any learnable parameters. Third, UNIREX (DLM-FP) and UNIREX (SLM-FP) dominate across all values, with AUPRC slowly decreasing as decreases. Even at $= 0.5$, they can still achieve high AUPRC scores of around 0.75. This suggests that UNIREX's gold rationale batching procedure (Sec. A.4) is helpful for learning from minimal gold rationale supervision, thus enabling effective plausibility optimization. In addition to these results on SST, see Fig. 8 for similar results on CoS-E.

4.5. Zero-Shot Faithfulness Transfer

In Table 1, we investigate if F_{ext}'s faithfulness, obtained via UNIREX training on some source (seen) dataset, can generalize to target (unseen) datasets/tasks in a zero-shot setting (*i.e.,* no fine-tuning on target datasets/tasks). In this experiment, we consider SST and sentiment analysis as the source dataset and task, respectively. We compare six methods: AA (IG), AA-F (Rand), UNIREX (AA-F), UNIREX (DLM-P), UNIREX (DLM-FP), and UNIREX (SLM-FP). For AA (IG), only F_{task} is trained on SST, since its F_{ext} is heuristic. Meanwhile, for the UNIREX variants, both F_{task} and F_{ext} are trained on SST. First, as an in-domain reference point, we report faithfulness and task performance on SST. Second, we evaluate on unseen target datasets for a seen task (*i.e.,* sentiment analysis): Yelp (Zhang et al., 2015) and Amazon (McAuley & Leskovec, 2013). Third, we evaluate

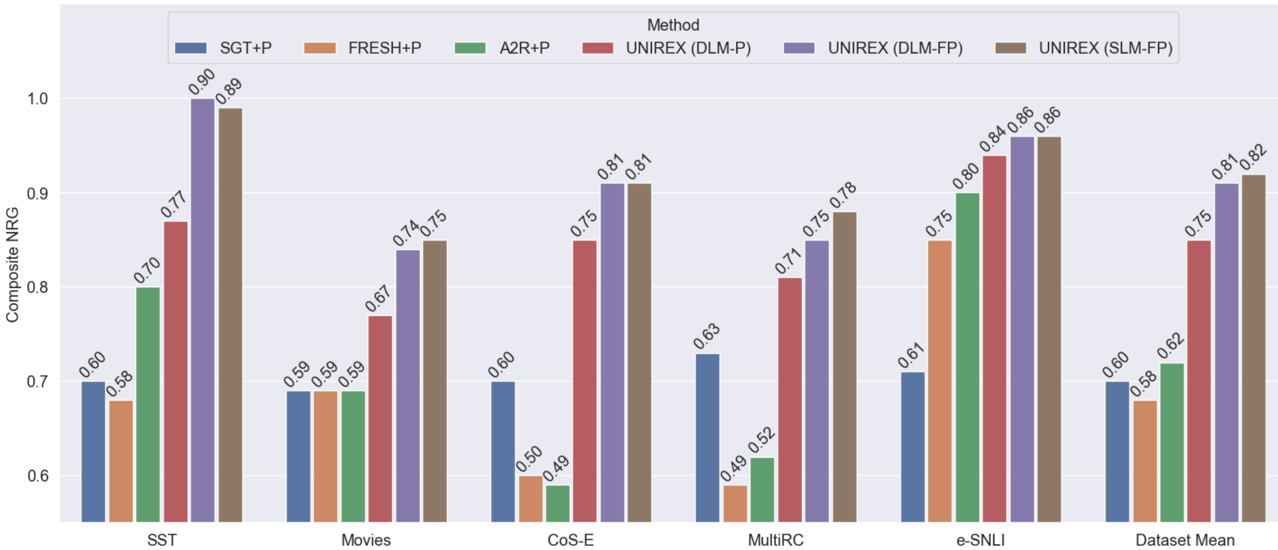

Figure 4: **Composite NRG Comparison (w/ Plausibility Optimization).** The composite NRG (CNRG) is the mean of the three desiderata NRG scores. For each dataset, we use CNRG to compare rationale extraction methods that *do* optimize for plausibility. Overall, UNIREX (SLM-FP) and UNIREX (DLM-FP) achieve the best CNRG – both beating the strongest baseline (*i.e.,* A2R+P) by over 30% on Dataset Mean – demonstrating UNIREX's ability to jointly optimize F_{task} and F_{ext} for all three desiderata.

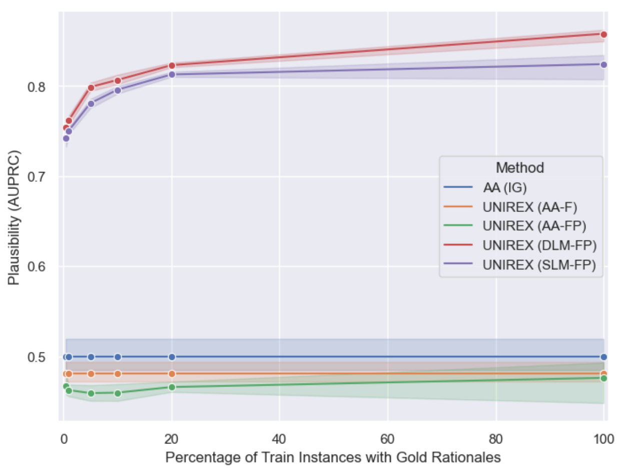

Figure 5: **Gold Rationale Efficiency on SST.**

on unseen target datasets for unseen target tasks: Stormfront (hate speech detection, binary F1) (de Gibert et al., 2018), OffenseEval (offensive speech detection, macro F1) (Zampieri et al., 2019), and SemEval2018 (irony detection, binary F1) (Van Hee et al., 2018).

We want to show that, even if F_{task} yields poor task performance on unseen datasets, F_{ext}'s rationales can still be faithful. As expected, all methods achieve much lower task performance in the third setting than in the first two settings. However, faithfulness does not appear to be strongly correlated with task performance, as unseen tasks' comp/suff scores are similar to seen tasks'. Across all datasets, DLM-FP has the best faithfulness and is the only method whose comp is always higher than suff. The other UNIREX variants are not as consistently strong as DLM-FP, but almost always beat non-UNIREX methods on comp and suff. Meanwhile, AA (IG) has the worst comp and suff overall. Ultimately, these results suggest that UNIREX-trained models' faithfulness (*i.e.,* alignment between F_{task}'s and F_{ext}'s outputs) is a dataset/task agnostic property (*i.e.,* can generalize across datasets/tasks), further establishing UNIREX's utility in low-resource settings.

References

Bastings, J. and Filippova, K. The elephant in the interpretability room: Why use attention as explanation when we have saliency methods? *arXiv preprint arXiv:2010.05607*, 2020.

Bastings, J., Aziz, W., and Titov, I. Interpretable neural predictions with differentiable binary variables. *arXiv preprint arXiv:1905.08160*, 2019.

Bender, E. M., Gebru, T., McMillan-Major, A., and Shmitchell, S. On the dangers of stochastic parrots: Can language models be too big?. In *Proceedings of the 2021 ACM Conference on Fairness, Accountability, and Transparency*, pp. 610–623, 2021.

Bhat, M. M., Sordoni, A., and Mukherjee, S. Self-training with few-shot rationalization: Teacher explanations aid student in few-shot nlu. *arXiv preprint arXiv:2109.08259*, 2021.

Camburu, O.-M., Rocktäschel, T., Lukasiewicz, T., and Blunsom, P. e-snli: Natural language inference with natural language explanations. *arXiv preprint arXiv:1812.01193*, 2018.

Carton, S., Rathore, A., and Tan, C. Evaluating and characterizing human rationales. *arXiv preprint arXiv:2010.04736*, 2020.

Chan, A., Xu, J., Long, B., Sanyal, S., Gupta, T., and Ren, X. Salkg: Learning from knowledge graph explanations for commonsense reasoning. *Advances in Neural Information Processing Systems*, 34, 2021.

de Gibert, O., Perez, N., García-Pablos, A., and Cuadros, M. Hate speech dataset from a white supremacy forum. *arXiv preprint arXiv:1809.04444*, 2018.

Denil, M., Demiraj, A., and De Freitas, N. Extraction of salient sentences from labelled documents. *arXiv preprint arXiv:1412.6815*, 2014.

Devlin, J., Chang, M.-W., Lee, K., and Toutanova, K. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

DeYoung, J., Jain, S., Rajani, N. F., Lehman, E., Xiong, C., Socher, R., and Wallace, B. C. Eraser: A benchmark to evaluate rationalized nlp models. *arXiv preprint arXiv:1911.03429*, 2019.

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

Falcon, W. and The PyTorch Lightning team. PyTorch Lightning, 3 2019. URL https://github.com/PyTorchLightning/pytorch-lightning.

Fleiss, J. L. Measuring nominal scale agreement among many raters. *Psychological bulletin*, 76(5):378, 1971.

Hase, P. and Bansal, M. Evaluating explainable ai: Which algorithmic explanations help users predict model behavior? *arXiv preprint arXiv:2005.01831*, 2020.

Hooker, S., Erhan, D., Kindermans, P.-J., and Kim, B. A benchmark for interpretability methods in deep neural networks. *arXiv preprint arXiv:1806.10758*, 2018.

Ismail, A. A., Corrada Bravo, H., and Feizi, S. Improving deep learning interpretability by saliency guided training. *Advances in Neural Information Processing Systems*, 34, 2021.

Jacovi, A. and Goldberg, Y. Towards faithfully interpretable nlp systems: How should we define and evaluate faithfulness? *arXiv preprint arXiv:2004.03685*, 2020.

Jain, S., Wiegreffe, S., Pinter, Y., and Wallace, B. C. Learning to faithfully rationalize by construction. *arXiv preprint arXiv:2005.00115*, 2020.

Kádár, A., Chrupała, G., and Alishahi, A. Representation of linguistic form and function in recurrent neural networks. *Computational Linguistics*, 43(4):761–780, 2017.

Khashabi, D., Chaturvedi, S., Roth, M., Upadhyay, S., and Roth, D. Looking beyond the surface: A challenge set for reading comprehension over multiple sentences. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pp. 252–262, 2018.

Kokhlikyan, N., Miglani, V., Martin, M., Wang, E., Alsallakh, B., Reynolds, J., Melnikov, A., Kliushkina, N., Araya, C., Yan, S., et al. Captum: A unified and generic model interpretability library for pytorch. *arXiv preprint arXiv:2009.07896*, 2020.

Kumar, S. and Talukdar, P. Nile: Natural language inference with faithful natural language explanations. *arXiv preprint arXiv:2005.12116*, 2020.

Lakhotia, K., Paranjape, B., Ghoshal, A., Yih, W.-t., Mehdad, Y., and Iyer, S. Fid-ex: Improving sequence-to-sequence models for extractive rationale generation. *arXiv preprint arXiv:2012.15482*, 2020.

Lei, T., Barzilay, R., and Jaakkola, T. Rationalizing neural predictions. *arXiv preprint arXiv:1606.04155*, 2016.

Li, J., Chen, X., Hovy, E., and Jurafsky, D. Visualizing and understanding neural models in nlp. *arXiv preprint arXiv:1506.01066*, 2015.

Li, J., Monroe, W., and Jurafsky, D. Understanding neural networks through representation erasure. *arXiv preprint arXiv:1612.08220*, 2016.

Lipton, Z. C. The mythos of model interpretability: In machine learning, the concept of interpretability is both important and slippery. *Queue*, 16(3):31–57, 2018.

Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., Levy, O., Lewis, M., Zettlemoyer, L., and Stoyanov, V. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*, 2019.

Lundberg, S. M. and Lee, S.-I. A unified approach to interpreting model predictions. In *Proceedings of the 31st international conference on neural information processing systems*, pp. 4768–4777, 2017.

Luo, S., Ivison, H., Han, C., and Poon, J. Local interpretations for explainable natural language processing: A survey. *arXiv preprint arXiv:2103.11072*, 2021.

McAuley, J. and Leskovec, J. Hidden factors and hidden topics: understanding rating dimensions with review text. In *Proceedings of the 7th ACM conference on Recommender systems*, pp. 165–172, 2013.

Minaee, S., Kalchbrenner, N., Cambria, E., Nikzad, N., Chenaghlu, M., and Gao, J. Deep learning–based text classification: A comprehensive review. *ACM Computing Surveys (CSUR)*, 54(3):1–40, 2021.

Narang, S., Raffel, C., Lee, K., Roberts, A., Fiedel, N., and Malkan, K. Wt5?! training text-to-text models to explain their predictions. *arXiv preprint arXiv:2004.14546*, 2020.

Paranjape, B., Joshi, M., Thickstun, J., Hajishirzi, H., and Zettlemoyer, L. An information bottleneck approach for controlling conciseness in rationale extraction. *arXiv preprint arXiv:2005.00652*, 2020.

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., et al. Pytorch: An imperative style, high-performance deep learning library. *Advances in neural information processing systems*, 32:8026–8037, 2019.

Poerner, N., Roth, B., and Schütze, H. Evaluating neural network explanation methods using hybrid documents and morphological agreement. *arXiv preprint arXiv:1801.06422*, 2018.

Pruthi, D., Dhingra, B., Soares, L. B., Collins, M., Lipton, Z. C., Neubig, G., and Cohen, W. W. Evaluating explanations: How much do explanations from the teacher aid students? *arXiv preprint arXiv:2012.00893*, 2020.

Rajani, N. F., McCann, B., Xiong, C., and Socher, R. Explain yourself! leveraging language models for commonsense reasoning. *arXiv preprint arXiv:1906.02361*, 2019.

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

Sanyal, S. and Ren, X. Discretized integrated gradients for explaining language models. *arXiv preprint arXiv:2108.13654*, 2021.

Schwarzenberg, R., Feldhus, N., and Möller, S. Efficient explanations from empirical explainers. *arXiv preprint arXiv:2103.15429*, 2021.

Shrikumar, A., Greenside, P., and Kundaje, A. Learning important features through propagating activation differences. In *International Conference on Machine Learning*, pp. 3145–3153. PMLR, 2017.

Simonyan, K., Vedaldi, A., and Zisserman, A. Deep inside convolutional networks: Visualising image classification models and saliency maps. *arXiv preprint arXiv:1312.6034*, 2013.

Situ, X., Zukerman, I., Paris, C., Maruf, S., and Haffari, G. Learning to explain: Generating stable explanations fast. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pp. 5340–5355, 2021.

Socher, R., Perelygin, A., Wu, J., Chuang, J., Manning, C. D., Ng, A. Y., and Potts, C. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the 2013 conference on empirical methods in natural language processing*, pp. 1631–1642, 2013.

Strout, J., Zhang, Y., and Mooney, R. J. Do human rationales improve machine explanations? *arXiv preprint arXiv:1905.13714*, 2019.

Sundararajan, M., Taly, A., and Yan, Q. Axiomatic attribution for deep networks. In *International Conference on Machine Learning*, pp. 3319–3328. PMLR, 2017.

Talmor, A., Herzig, J., Lourie, N., and Berant, J. Commonsenseqa: A question answering challenge targeting commonsense knowledge. *arXiv preprint arXiv:1811.00937*, 2018.

Van Hee, C., Lefever, E., and Hoste, V. Semeval-2018 task 3: Irony detection in english tweets. In *Proceedings of The 12th International Workshop on Semantic Evaluation*, pp. 39–50, 2018.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. Attention is all you need. In *Advances in neural information processing systems*, pp. 5998–6008, 2017.

Ye, Q., Lin, B. Y., and Ren, X. Crossfit: A few-shot learning challenge for cross-task generalization in nlp. *arXiv preprint arXiv:2104.08835*, 2021.

Yu, M., Chang, S., Zhang, Y., and Jaakkola, T. S. Rethinking cooperative rationalization: Introspective extraction and complement control. *arXiv preprint arXiv:1910.13294*, 2019.

Yu, M., Zhang, Y., Chang, S., and Jaakkola, T. Understanding interlocking dynamics of cooperative rationalization. *Advances in Neural Information Processing Systems*, 34, 2021.

Zaheer, M., Guruganesh, G., Dubey, K. A., Ainslie, J., Alberti, C., Ontanon, S., Pham, P., Ravula, A., Wang, Q., Yang, L., et al. Big bird: Transformers for longer sequences. In *NeurIPS*, 2020.

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

Zampieri, M., Malmasi, S., Nakov, P., Rosenthal, S., Farra, N., and Kumar, R. Semeval-2019 task 6: Identifying and categorizing offensive language in social media (offenseval). *arXiv preprint arXiv:1903.08983*, 2019.

Zhang, X., Zhao, J., and LeCun, Y. Character-level Convolutional Networks for Text Classification. *arXiv:1509.01626 [cs]*, September 2015.

A. Appendix

A.1. Related Work

Faithfulness Many prior works have tried to improve the faithfulness of extractive rationales through the use of AAs (Bastings & Filippova, 2020). Typically, this involves designing gradient-based (Sundararajan et al., 2017; Denil et al., 2014; Lundberg & Lee, 2017; Li et al., 2015) or perturbation-based (Li et al., 2016; Poerner et al., 2018; Kádár et al., 2017) AAs. However, attribution algorithms cannot be optimized and tend to be compute-intensive (often requiring multiple LM forward/backward passes). Recently, Ismail et al. (2021) addressed the optimization issue by regularizing the task model to yield faithful rationales via the AA, while other works (Situ et al., 2021; Schwarzenberg et al., 2021) addressed the compute cost issue by training an LM (requiring only one forward pass) to mimic an AA's behavior. Another line of work aims to produce faithful rationales by construction, via SPPs (Jain et al., 2020; Yu et al., 2021; Paranjape et al., 2020; Bastings et al., 2019; Yu et al., 2019; Lei et al., 2016). Still, SPPs' faithfulness can only guarantee sufficiency – not comprehensiveness (DeYoung et al., 2019). Also, SPPs generally perform worse than vanilla LMs because they hide much of the original text input from the predictor and are hard to train end-to-end.

Plausibility Existing approaches for improving extractive rationale plausibility typically involve supervising LM-based extractors (Bhat et al., 2021) or SPPs (Jain et al., 2020; Paranjape et al., 2020; DeYoung et al., 2019) with gold rationales. However, existing LM-based extractors have not been trained for faithfulness, while SPPs' faithfulness by construction comes at the great cost of task performance. Meanwhile, more existing works focus on improving the plausibility of free-text rationales (Narang et al., 2020; Lakhotia et al., 2020; Camburu et al., 2018), often with task-specific pipelines (Rajani et al., 2019; Kumar & Talukdar, 2020).

Connection to UNIREX Unlike prior works, UNIREX enables both the task model and rationale extractor to be jointly optimized for faithfulness, plausibility, and task performance. As a result, UNIREX-trained rationale extractors achieve a better balance of faithfulness and plausibility, without compromising the task model's performance. Also, by using a learned rationale extractor, which generally only requires one model forward pass, UNIREX does not have the computational expenses that limit many AAs.

A.2. Text Classification

Here, we formalize the text classification problem in more detail. Let $D = \{X, Y\}_{i=1}^{N}$ be a dataset, where $X = \{\mathbf{x}_i\}_{i=1}^{N}$ are the text inputs, $Y = \{y_i\}_{i=1}^{N}$ are the labels, and N is the number of instances $(\mathbf{x}_i, y_i)$ in D. We also as-

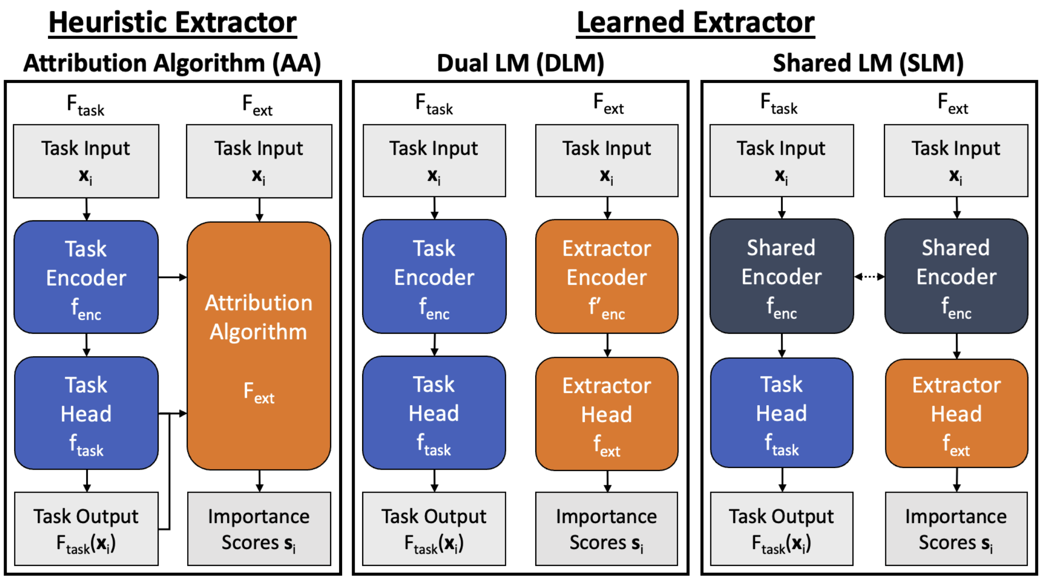

Figure 6: **Rationale Extractor Types.** In general, rationale extractor F_{ext} can be either heuristic or learned. A heuristic F_{ext} is a handcrafted **attribution algorithm**, which cannot be trained (Sec. 3.2.1). By default, UNIREX uses a learned F_{ext}, which can be optimized for faithfulness, plausibility, and task performance. For learned F_{ext}, we focus on two architectures (w.r.t. task model F_{task}): **Dual LM** designs F_{task} and F_{ext} as two fully separate LMs, while **Shared LM** designs F_{task} and F_{ext} to share the same text encoder (Sec. 3.2.2). Although some operations within UNIREX are non-differentiable, Shared LM's shared encoder allows us to approximate end-to-end training of both models w.r.t. all three desiderata (Sec. 3.4, Fig. 2).

sume D can be partitioned into train set D_{train}, dev set D_{dev}, and test set D_{test}. Let $F_{\text{task}} = f_{\text{task}}(f_{\text{enc}}(\cdot))$ be a task LM, where f_{enc} is the text encoder, and f_{task} is the task output head. Typically, F_{task} has a BERT-style architecture (Devlin et al., 2018), in which f_{enc} is a Transformer (Vaswani et al., 2017) while f_{task} is a linear layer. Below, we define the sequence classification (SST, Movies, MultiRC, e-SNLI) and multi-choice QA (CoS-E) tasks, which are different types of text classification.

Sequence Classification In sequence classification, $\mathbf{x}_i$ is a token sequence (*e.g.,* a single sentence, a pair of sentences), while y_i is the target class for $\mathbf{x}_i$. Here, we assume a fixed label space $Y = \{1, \dots, M\}$ of size M, where $y_i \in Y$ for all i. Thus, f_{task} outputs a vector of size M, such that $F_{\text{task}}(\mathbf{x}_i) = f_{\text{task}}(f_{\text{enc}}(\mathbf{x}_i)) = \hat{\mathbf{y}}_i \in \mathbb{R}^M$ is the logit vector used to classify $\mathbf{x}_i$. Given $\hat{\mathbf{y}}_i = [\hat{y}_{i,j}]_{j=1}^M$, let $\tilde{y}_i = \arg\max_j \hat{y}_{i,j}$ be the class predicted by F_{task}. The goal of sequence classification is to learn F_{task} such that $\tilde{y}_i = y_i$, for all $(\mathbf{x}_i, y_i)$ (Minaee et al., 2021).

Multi-Choice QA Instead of a fixed label space, multi-choice QA has a different (but fixed-size) set of answer choices per instance. For instance i, let q_i be the question (*e.g., "A friend is greeting me, what would they say?"*) and $A_i = \{a_{i,j}\}_{j=1}^M$ be the corresponding answer choices (*e.g.,* {*"say hello"*, *"greet"*, *"associate"*, *"socialize"*, *"smile"*}), where M is now the number of answer choices. Define $\mathbf{x}_{i,j} = q_i \oplus a_{i,j}$, where $\oplus$ denotes concatenation. In multi-choice QA, we have $\mathbf{x}_i = \{\mathbf{x}_{i,j}\}_{j=1}^M$, while $y_i \in A_i$ is the correct answer for $\mathbf{x}_i$. Thus, f_{task} outputs a scalar, such that $F_{\text{task}}(\mathbf{x}_{i,j}) = f_{\text{task}}(f_{\text{enc}}(\mathbf{x}_{i,j})) = \hat{y}_{i,j} \in \mathbb{R}$ is the logit for $\mathbf{x}_{i,j}$. Given $\hat{\mathbf{y}}_i = [\hat{y}_{i,j}]_{j=1}^M$, let $j' = \arg\max_j \hat{y}_{i,j}$, where $\tilde{y}_i = a_{i,j'}$ is the answer predicted by F_{task}. The goal of multi-choice QA is to learn F_{task} such that $\tilde{y}_i = y_i$, for all $(\mathbf{x}_i, y_i)$ (Talmor et al., 2018).

A.3. Ablation Studies

We present five ablation studies to validate the effectiveness of our UNIREX design choices. The results of these ablation studies are displayed in Table 2, where each of the five sections contains results for a different ablation. Thus, all numbers within the same section (ablation) and column (metric) are comparable.

Extractor Type (F) In the Ext Type (F) section, we compare four heuristic rationale extractors, using AA-F. In this case, besides task performance, we can only optimize (the task model) for faithfulness. Rand uses random importance scores, Gold directly uses the gold rationales, Inv uses the inverse of the gold rationales, and IG uses IG rationales. All heuristics yield similar task performance, but IG dominates on all faithfulness metrics. This makes sense because IG is computed using F_{task}'s inputs/parameters/outputs, while the others do not have this information. For plausibility, Gold

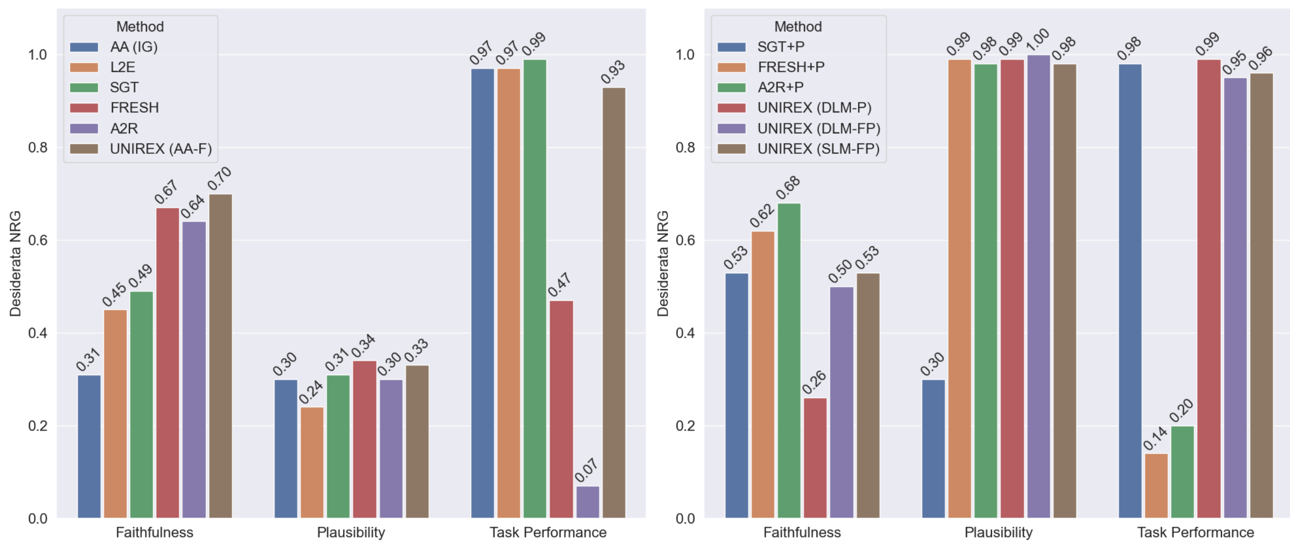

Figure 7: **Desiderata NRG Comparison.** For each rationale extraction method, we show the desiderata NRG for faithfulness (FNRG), plausibility (PNRG), and task performance (TNRG), averaged over all datasets. **Left**: This plot compares methods *without* plausibility optimization. UNIREX (AA-F)'s FNRG is highest, while its TNRG is close to highest. Meanwhile, baselines with high FNRG (*i.e.,* FRESH, A2R) have low TNRG, while baselines with high TNRG (*i.e.,* AA (IG), L2E, SGT) have low FNRG. **Right**: This plot compares methods *with* plausibility optimization. UNIREX (DLM-FP) and UNIREX (SLM-FP) have moderate FNRG, but the highest (or near-highest) PNRG and TNRG. Meanwhile, baselines with high FNRG (*i.e.,* FRESH+P, A2R+P) have low TNRG, while baselines with high TNRG (*i.e.,* SGT+P) have low PNRG.

is the best, Inv is the worst, and Rand and IG are about the same, as none of the heuristics are optimized for plausibility.

Extractor Type (P) In the Ext Type (FP) section, we compare four learned rationale extractors. In this case, besides task performance, we can optimize for both faithfulness and plausibility. By default, attribution algorithms' dimension scores are pooled into token scores via sum pooling. AA-FP (Sum) uses IG with sum pooling, while AA-FP (MLP) replaces the sum pooler with a MLP-based pooler to increase capacity for plausibility optimization. Task performance for all four methods is similar, AA-FP (Sum) dominates on faithfulness, and DLM-FP and SLM-FP dominate on plausibility. AA-FP (MLP) does not perform as well on faithfulness but slightly improves on plausibility compared to AA-FP (Sum).

Comp/Suff Losses The Comp/Suff Loss section compares different combinations of Comp and Suff losses, using SLM-FP. Note that SLM-FP (Comp+Suff) is equivalent to SLM-FP shown in other tables/sections. As expected, SLM-FP (Comp) does best on Comp, but SLM-FP (Comp+Suff) actually does best on Suff. Meanwhile, SLM-FP, (Suff) does second-best on Suff but is much worse on Comp. This shows that Comp and Suff are complementary for optimization.

Suff Criterion The Suff Criterion section compares different Suff criteria, using SLM-FP. SLM-FP (KLDiv) uses the KL divergence criterion, SLM-FP (MAE) uses the MAE criterion, and SLM-FP (Margin) uses the margin criterion. SLM-FP (Margin) is equivalent to SLM-FP in other tables/sections. All criteria yield similar performance and plausibility, while Margin is slightly better on faithfulness.

SLM Extractor Head The SLM Ext Head section compares different extractor heads, using SLM-FP. Linear is the default choice and uses a linear layer. MLP-2048-2 uses a MLP with two 2048-dim hidden layers. MLP-4096-3 uses a MLP with three 4096-dim hidden layers. All three output head types yield similar performance, but decreasing head capacity yields better faithfulness, while increasing head capacity heads yields better plausibility. This trades off faithfulness and plausibility, although larger heads will be more compute-intensive.

A.4. Gold Rationale Supervision

If a learned rationale extractor is chosen, UNIREX enables users to specify how much gold rationale supervision to use. Ideally, each train instance would be annotated with a gold rationale. In this case, we could directly minimize the plausibility loss for each train instance. However, since gold rationales can be expensive to annotate, UNIREX provides a special batching procedure for training with limited gold rationale supervision.

Given $N_{\text{train}} = |D_{\text{train}}|$ train instances, let $0 < \quad < 100$ be the percentage of train instances with gold rationales, $N_{\text{gold}} = \lceil \frac{}{100} N_{\text{train}} \rceil \quad 1$ be the number of train instances with gold rationales, b be the desired train batch size, and

$\gamma > 1$ be a scaling factor. Define $D_{\text{gold}} \subseteq D_{\text{train}}$ as the set of train instances with gold rationales, where $|D_{\text{gold}}| = N_{\text{gold}}$. Note that, if all train instances have gold rationales, then $D_{\text{gold}} = D_{\text{train}}$ and $\gamma = 100$.

Each batch is constructed as follows: (1) randomly sample $b_{\text{gold}} = \max(1, \frac{b}{\gamma})$ instances from D_{gold} without replacement, then (2) randomly sample $b - b_{\text{gold}}$ instances from $D_{\text{train}} \setminus D_{\text{gold}}$ without replacement. This results in a batch with b total train instances, b_{gold} with gold rationales and the rest without. Since N_{gold} is generally small, we only sample from D_{gold} without replacement for a given batch, but not a given epoch. Thus, instances from D_{gold} may appear more than once in the same epoch. However, we do sample from $D_{\text{train}} \setminus D_{\text{gold}}$ without replacement for each batch and epoch, so every instance in $D_{\text{train}} \setminus D_{\text{gold}}$ appears exactly once per epoch.

After constructing the batch, we compute the plausibility loss for the batch as follows: $\sum_{i=1}^{b} \mathbb{1}_{(\mathbf{x}_i, y_i) \in D_{\text{gold}}} L_{\text{plaus}}(F_{\text{ext}}(\mathbf{x}_i), \mathbf{r}_i)$, where L_{plaus} is the plausibility loss for train instance $(\mathbf{x}_i, y_i)$. This function zeroes out the plausibility loss for instances without gold rationales, so that plausibility is only being optimized with respect to instances with gold rationales. However, in Sec. 4.4, we show that it is possible to achieve high plausibility via rationale extractors trained on minimal gold rationale supervision.

A.5. Plausibility User Study

Gold rationale based plausibility evaluation is noisy because gold rationales are for the target label, not a F_{task}'s predicted label. Thus, we conduct two five-annotator user studies (Table 3) to get a better plausibility measurement. Given 50 random test instances from SST, we get the rationales for SGT+P, A2R+P, UNIREX (AA-FP), and UNIREX (DLM-FP), plus the gold rationales. For each instance, we threshold all rationales to have the same number of positive tokens as the gold rationale. The first user study is forward simulation (Hase & Bansal, 2020; Jain et al., 2020). Here, the annotator is given an input and a rationale for some model's prediction, then asked what (binary) sentiment label the model most likely predicted. For forward simulation, we also consider a No Rationale baseline, where no tokens are highlighted. For No Rationale and Gold (which we call "oracle methods"), the target label is the correct choice. Annotators are also asked to rate their confidence (4-point Likert scale) in their answer to this question. The second user study involves giving a subjective rating of how plausible the rationale is (Hase & Bansal, 2020). Here, the annotator is given the input, rationale, and model's predicted label, then asked to rate (5-point Likert scale) how aligned the rationale is with the prediction.

In both accuracy and subjective rating, we find that DLM-FP performs best among all non-oracle methods and even slightly beats Gold on accuracy, further supporting our claim that DLM-FP rationales are plausible. As expected, the fact that Gold does not achieve near-100% accuracy shows the discrepancy between evaluating plausibility based on the target label (*i.e.,* gold rationale similarity) and F_{task}'s predicted label (forward simulation). Meanwhile, SGT+P and AA-FP, which had lower AUPRC and TF1 in our automatic evaluation, also do worse in accuracy and alignment. Also, users found SGT+P and AA-FP rationales harder to understand, as shown by their lower confidence scores. Meanwhile, A2R+P had high AUPRC and TF1, but gets very low accuracy and alignment because A2R+P's predicted label was often not the target label, leading to misalignment with its gold-like rationale. Nonetheless, users were still most confident in their predictions using A2R+P's rationales. A2R+P is a great example of how automatic plausibility evaluation can be misleading. For the accuracy, confidence, and alignment questions, we achieved Fleiss' Kappa (Fleiss, 1971) inter-annotator agreement scores of 0.2456 (fair), 0.1282 (slight), and, 0.1561 (slight), respectively. This lack of agreement demonstrates the difficulty of measuring rationale plausibility.

A.6. Explainability Objectives

A.6.1. Faithfulness

Sufficiency In addition, to the criteria presented in Sec. 3.3, we consider two other sufficiency loss functions. The first is the *KL divergence criterion* used in (Ismail et al., 2021), which considers the entire label distribution and is defined as $L_{\text{suff-KL}} = \text{KL}(F_{\text{task}}(\mathbf{r}_i^{(k)}) \,||\, F_{\text{task}}(\mathbf{x}_i))$. The second is the *mean absolute error (MAE) criterion*, which is defined as $L_{\text{suff-MAE}} = |L_{\text{CE}}(F_{\text{task}}(\mathbf{r}_i^{(k)}), y_i) - L_{\text{CE}}(F_{\text{task}}(\mathbf{x}_i), y_i)|$. Unlike the difference criterion $L_{\text{suff-diff}}$ and margin criterion $L_{\text{suff-margin}}$ (Sec. 3.3), the MAE criterion assumes that using $\mathbf{r}_i^{(k)}$ as input should not yield better task performance than using $\mathbf{x}_i$ as input. In our experiments, we find that $L_{\text{suff-margin}}$ is effective, though others (*e.g.,* KL divergence, MAE) can be used too.

A.6.2. Plausibility

Similar to faithfulness, UNIREX places no restrictions on the choice of plausibility objective. As described in Sec. 3.3, given gold rationale $\mathbf{r}_i$ for input $\mathbf{x}_i$, plausibility optimization entails training F_{ext} to predict binary importance label $\mathbf{r}_i^{\,t}$ for each token x_i^t. This is essentially binary token classification, so one natural choice for L_{plaus} is the token-level binary cross-entropy (BCE) criterion: $L_{\text{plaus-BCE}} = -\sum_t \mathbf{r}_i^{\,t} \log(F_{\text{ext}}(x_i^t))$ (Sec. 3.3). Another option is the sequence-level *KL divergence criterion*, which is defined as: $L_{\text{plaus-KL}} = \text{KL}(F_{\text{ext}}(\mathbf{x}_i) \,||\, \mathbf{r}_i)$.

Additionally, we can directly penalize $F_{\text{ext}}(\mathbf{x}_i)$ in the

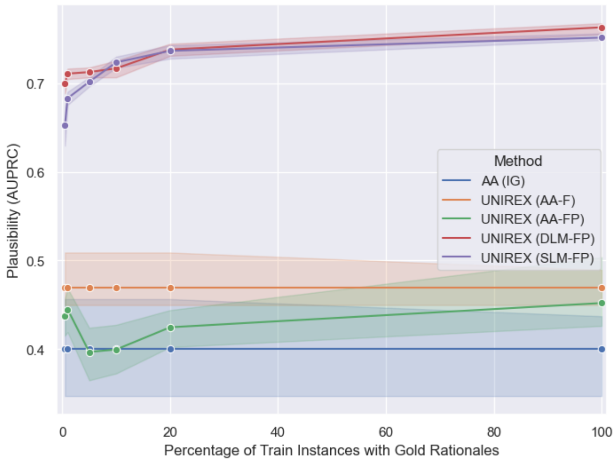

Figure 8: **Gold Rationale Efficiency on CoS-E.**

logit space via a *linear loss*, defined as: $L_{\text{plaus-linear}} = \phi(\mathbf{r}_i)\, F_{\text{ext}}(\mathbf{x}_i)$, where $\phi(u) = -2u + 1$ maps positive and negative tokens to -1 and $+1$, respectively. The linear loss directly pushes the logits corresponding to positive/negative tokens to be higher/lower and increase the margin between them. To prevent linear loss values from becoming arbitrarily negative, we can also lower bound the loss with a margin m_p, yielding: $L_{\text{plaus-linear-margin}} = \max(-m_p, L_{\text{plaus-linear}}) + m_p$.

A.7. Implementation Details

For the LM architecture of F_{task} and F_{ext}, we use BigBird-Base (Zaheer et al., 2020) in all our experiments, in order to handle input sequences with up to 4096 tokens. For all AA-based methods besides vanilla AA, we use the IG (Sundararajan et al., 2017) AA, which has been commonly adopted in the explainability literature (Pruthi et al., 2020; Sanyal & Ren, 2021; Ismail et al., 2021). By default, IG requires 50 steps (*i.e.,* backward passes) per instance (Kokhlikyan et al., 2020), which is prohibitive when regularizing the LM via IG, so we use 3 steps in both training and evaluation. For all experiments, we use a learning rate of $2e-5$ and effective batch size of 32. We train for a maximum of 10 epochs, with early stopping patience of 5 epochs. We only tune faithfulness and plausibility loss weights, sweeping $\lambda_c = [0.5, 0.7, 1.0]$, $\lambda_s = [0.5, 0.7, 1.0]$, and $\lambda_p = [0.5, 0.7, 1.0]$. We find that $\lambda_c = 0.5$ and $\lambda_s = 0.5$ are usually best. For each method variant, we tuned hyperparameters w.r.t. dev CNRG, computed across all hyperparameter configurations for the variant. For the batching factor β (Sec. A.4), we use 2. All experiments are implemented using PyTorch-Lightning (Paszke et al., 2019; Falcon & The PyTorch Lightning team, 2019).

A.8. Gold Rationale Efficiency

Fig. 8 shows the gold rationale data efficiency results for CoS-E, using the same setup as Sec. 4.4. Overall, we see that the CoS-E results are quite similar to the SST results. Again, UNIREX (DLM-FP) and UNIREX (SLM-FP) dominate across all γ values, with AUPRC slowly decreasing as γ decreases. Interestingly, UNIREX (AA-FP) yields a noticeable dip in AUPRC for lower γ values. Since AA-FP has limited capacity (via the task model) for plausibility optimization, it is possible that this fluctuation is due to random noise. We leave further analysis of this for future work.

A.9. Compute Efficiency

Besides faithfulness, plausibility, and task performance, compute efficiency is also an important desideratum of rationale extraction. With that in mind, the number of IG steps is a critical design choice. A higher number of IG steps means a more accurate IG approximation, which should may yield more faithful rationales. On the other hand, a higher number of IG steps means greater computational costs. As stated in Sec. A.7, we use 3-step IG in all of our experiments, in order to make all methods computationally comparable. 3-step IG requires three backward passes, while all other compared methods require either one forward pass or one backward pass. We would like to empirically characterize this trade-off between faithfulness and compute efficiency. Thus, we compare IG performance for $\zeta = [3, 5, 10, 30, 50, 70, 100, 500, 1000]$ steps, denoted as AA (IG-ζ). In Table 4, we report faithfulness, plausibility, task performance, convergence delta (*i.e.,* IG approximation error), and inference time for each IG-based method. Additionally, we compare these IG settings to UNIREX (AA-F) (which uses 3-step IG), UNIREX (DLM-FP), and UNIREX (SLM-FP). Since UNIREX (DLM-FP) and UNIREX (SLM-FP) are not IG-based, we do not report convergence delta for them.

As expected, we find that convergence delta decreases as ζ increases. Furthermore, AA (IG) faithfulness scores generally improve as ζ increases, although the improvement begins to saturate at around $\zeta = 50$. Similarly, we find that AA (IG) plausibility scores for IG also tend to improve as ζ increases, with the improvement also saturating around $\zeta = 50$. This makes sense because plausibility optimization involves regularizing the task model to yield rationales that are similar to gold rationales, but this regularization is less effective if the yielded rationales are less faithful to the task model. However, despite these faithfulness and plausibility improvements, we see that inference time increases roughly linearly w.r.t. ζ. In particular, AA (IG-1000) is over 200 times slower than AA (IG-3). Meanwhile, despite only using 3-step IG, UNIREX (AA-F) beats all AA (IG) variants on faithfulness, while achieving similar plausibility to AA (IG-3). On the other hand, UNIREX (DLM-FP) and UNIREX (SLM-FP) achieve the best faithfulness, while also achieving the best plausibility and inference time by far, without sacrificing task performance. Both UNIREX (DLM-FP) and UNIREX (SLM-FP) are over 2000 times faster than AA (IG-1000) and over nine times faster than AA (IG-3). These results show that UNIREX (esp. using learned rationale extractor) is an effective framework for jointly optimizing rationale extraction for faithfulness, plausibility, and task performance, while also achieving high compute efficiency.

A.10. Qualitative Analysis

In Sec. A.5, we presented a user study of fifty SST test instances to further evaluate the plausibility of rationales produced by various methods. To get additional insights about rationale plausibility, we conduct qualitative analysis of selected instances from the user study. In Table 5, for each selected instance and a given method, we show the method's rationale, F_{task}'s corresponding prediction, and the user-annotated alignment score for the rationale w.r.t. the prediction. We consider two groups of selected instances.

First, we select the top-3 instances w.r.t. mean alignment score (across five annotators), averaged over all methods (*i.e.,* Instances 30, 41, and 35). Not surprisingly, we find that F_{task}'s prediction is very consistent across all methods. For these three instances, all of the predicted labels happen to be both positive and correct (*i.e.,* same as gold label). Similarly, the rationales are also rather consistent across methods. In particular, A2R+P, UNIREX (DLM-FP), and Gold consistently yield the highest alignment scores. In Instance 30, these three methods plausibly highlight "good" and "actress". In Instance 41, these three methods plausibly highlight "understands", "medium", "amazingly", and "well". In Instance 35, these three methods plausibly highlight "admirable" and "achievement". Of course, this is expected for Gold, since gold rationales are human-annotated w.r.t. the gold label. The high alignment scores for A2R+P and UNIREX (DLM-FP) also make sense since A2R+P and UNIREX (DLM-FP) yielded high PNRG (Fig. 7).

Second, we select the top-3 instances w.r.t. standard deviation (std) alignment score (across five annotators), averaged over all methods (*i.e.,* Instances 9, 12, 19). This time, there is greater variance in the per-instance alignment scores across methods. For these three instances, the negative label is predicted by all methods except A2R+P. Like before, UNIREX (DLM-FP) and Gold consistently yield the highest alignment scores. In Instance 9, these two methods plausibly highlight "bad" and "ridiculous". In Instance 12, these two methods plausibly highlight "avoid". In Instance 19, these two methods plausibly highlight "lameness" and "bad". However, this time, A2R+P consistently yields the lowest alignment scores. Even though A2R+P produces rationales that are similar to those of UNIREX (DLM-FP) and Gold (*i.e.,* aligning with the gold label), A2R+P's rationales do not support A2R+P's predicted label. This illustrates the limitation of automatically evaluating plausibility via gold rationale similarity, as A2R+P achieved high PNRG. Meanwhile, we see that UNIREX (DLM-FP) consistently yields the highest plausibility across various types of evaluation (*i.e.,* gold rationale similarity, forward simulation, subjective rating).

A.11. Additional Empirical Results

In this subsection, we present additional results from our experiments. Besides the aggregated results shown in Sec. 4 of the main text, Tables 6-12 contain more detailed results, using both raw and NRG metrics. Specifically, Tables 6-10 show all raw/NRG results for each dataset, Table 11 shows the ablation results for all raw metrics, and Table 12 includes the zero-shot explainability transfer results for UNIREX (SLM-FP). Generally, the computation of NRG should involve globally aggregating the raw metrics for all available methods, as done in the main results. However, for a number of more focused experiments (Tables 11-12), only a subset of the available methods are considered. Thus, to make the faithfulness results in Tables 11-12 easier to digest, we introduce a metric called Comp-Suff Difference (CSD), which locally aggregates comp and suff as: CSD = comp suff. Therefore, since higher/lower comp/suff signal higher faithfulness, then higher CSD signals higher faithfulness.

Task	Dataset	Method	Faithfulness			Task Performance
			CSD (″)	Comp (″)	Suff (#)	Perf (″)
SA	SST	AA (IG)	-0.138 (0.040)	0.119 (0.009)	0.258 (0.031)	93.81 (0.55)
		AA-F (Rand)	-0.156 (0.018)	0.171 (0.040)	0.327 (0.050)	94.05 (0.35)
		UNIREX (AA-F)	0.120 (0.055)	0.292 (0.051)	0.171 (0.038)	92.97 (0.44)
		UNIREX (DLM-P)	-0.113 (0.040)	0.142 (0.008)	0.255 (0.007)	**94.86** (0.41)
		UNIREX (DLM-FP)	0.151 (0.056)	**0.319** (0.090)	0.167 (0.036)	93.81 (0.54)
		UNIREX (SLM-FP)	**0.189** (0.030)	0.302 (0.039)	**0.113** (0.013)	93.68 (0.67)
	Yelp	AA (IG)	-0.149 (0.028)	0.069 (0.004)	0.219 (0.028)	92.50 (2.07)
		AA-F (Rand)	-0.177 (0.056)	0.127 (0.022)	0.305 (0.060)	86.27 (7.88)
		UNIREX (AA-F)	0.013 (0.036)	0.138 (0.078)	0.126 (0.059)	83.93 (13.20)
		UNIREX (DLM-P)	-0.004 (0.028)	0.138 (0.009)	0.143 (0.023)	**93.33** (0.90)
		UNIREX (DLM-FP)	**0.169** (0.060)	**0.265** (0.094)	0.097 (0.033)	92.37 (0.46)
		UNIREX (SLM-FP)	0.114 (0.056)	0.175 (0.055)	**0.060** (0.001)	86.60 (1.57)
	Amazon	AA (IG)	-0.148 (0.038)	0.076 (0.010)	0.224 (0.037)	**91.13** (0.28)
		AA-F (Rand)	-0.166 (0.035)	0.120 (0.021)	0.286 (0.035)	81.52 (6.95)
		UNIREX (AA-F)	0.057 (0.106)	0.130 (0.077)	0.073 (0.039)	77.90 (13.12)
		UNIREX (DLM-P)	-0.017 (0.027)	0.142 (0.001)	0.158 (0.026)	90.92 (0.93)
		UNIREX (DLM-FP)	**0.133** (0.039)	**0.232** (0.072)	0.098 (0.033)	89.35 (2.22)
		UNIREX (SLM-FP)	0.097 (0.027)	0.147 (0.012)	**0.050** (0.017)	81.82 (7.62)
HSD	Stormfront	AA (IG)	-0.109 (0.053)	0.135 (0.010)	0.245 (0.059)	10.48 (1.66)
		AA-F (Rand)	-0.147 (0.021)	0.150 (0.020)	0.297 (0.005)	**10.66** (2.86)
		UNIREX (AA-F)	**0.127** (0.015)	**0.219** (0.009)	0.092 (0.025)	10.36 (1.94)
		UNIREX (DLM-P)	-0.125 (0.092)	0.122 (0.008)	0.246 (0.099)	10.10 (1.73)
		UNIREX (DLM-FP)	0.052 (0.027)	0.167 (0.084)	0.115 (0.059)	10.37 (2.66)
		UNIREX (SLM-FP)	0.049 (0.041)	0.110 (0.039)	**0.062** (0.043)	4.51 (1.87)
OSD	OffenseEval	AA (IG)	-0.146 (0.044)	0.097 (0.009)	0.244 (0.052)	33.51 (0.99)
		AA (Rand)	-0.148 (0.046)	0.101 (0.020)	0.249 (0.065)	34.08 (2.34)
		UNIREX (AA-F)	-0.029 (0.040)	0.074 (0.040)	0.102 (0.024)	32.62 (4.85)
		UNIREX (DLM-P)	-0.102 (0.073)	0.112 (0.010)	0.214 (0.081)	33.67 (1.01)
		UNIREX (DLM-FP)	**0.053** (0.012)	**0.140** (0.049)	0.087 (0.045)	35.52 (1.26)
		UNIREX (SLM-FP)	0.039 (0.031)	0.087 (0.016)	**0.048** (0.024)	**38.17** (0.96)
ID	SemEval2018	AA (IG)	-0.120 (0.061)	0.128 (0.014)	0.248 (0.064)	29.63 (4.72)
		AA-F (Rand)	-0.133 (0.043)	0.124 (0.013)	0.258 (0.053)	32.39 (9.73)
		UNIREX (AA-F)	-0.028 (0.051)	0.069 (0.041)	0.096 (0.011)	**49.95** (8.31)
		UNIREX (DLM-P)	-0.112 (0.095)	0.140 (0.017)	0.252 (0.112)	27.78 (5.08)
		UNIREX (DLM-FP)	**0.047** (0.017)	**0.149** (0.052)	0.102 (0.053)	31.97 (2.80)
		UNIREX (SLM-FP)	0.027 (0.047)	0.091 (0.027)	**0.064** (0.033)	17.42 (4.04)

Table 1: **Zero-Shot Faithfulness Transfer from SST.** We investigate whether the faithfulness of UNIREX rationale extractors (AA-F, DLM-FP) trained on SST can generalize to unseen datasets/tasks, even when the task model's task performance cannot. Also, we include AA (IG) as a heuristic extractor baseline (*i.e.,* only the task model is trained). Here, the seen task is sentiment analysis (SA), while the unseen tasks are hate speech detection (HSD), offensive speech detection (OSD), and irony detection (ID). For SA, the unseen datasets are Yelp and Amazon. For HSD, OSD, and ID, the unseen datasets are Stormfront, OffenseEval, and SemEval2018, respectively. Overall, we find that faithfulness is not strongly correlated with task performance, as unseen tasks' comp/suff scores are similar to seen tasks'. In particular, though all methods achieve poor task performance on unseen tasks, UNIREX (DLM-FP)'s comp/suff scores are consistently good across all tasks, demonstrating its faithfulness generalization ability.

Ablation	UNIREX Config	Faithfulness		Plausibility	Performance
		Comp (″)	Suff (#)	AUPRC (″)	Acc (″)
Ext Type (F)	AA-F (Rand)	0.171 (0.040)	0.327 (0.050)	44.92 (0.00)	**94.05** (0.35)
	AA-F (Gold)	0.232 (0.088)	0.249 (0.021)	**100.00** (0.00)	93.81 (0.54)
	AA-F (Inv)	0.242 (0.010)	0.357 (0.019)	20.49 (0.00)	93.47 (1.81)
	AA-F (IG)	**0.292** (0.051)	**0.171** (0.038)	48.13 (1.14)	92.97 (0.44)
Ext Type (FP)	AA-FP (Sum)	0.296 (0.067)	0.185 (0.048)	47.60 (2.44)	93.25 (0.45)
	AA-FP (MLP)	0.285 (0.051)	0.197 (0.100)	54.82 (1.97)	93.23 (0.92)
	DLM-FP	**0.319** (0.090)	0.167 (0.036)	**85.80** (0.74)	**93.81** (0.18)
	SLM-FP	0.302 (0.039)	**0.113** (0.013)	82.55 (0.84)	93.68 (0.67)
Comp/Suff Loss	SLM-FP (Comp)	**0.350** (0.048)	0.310 (0.049)	82.79 (0.62)	93.59 (0.11)
	SLM-FP (Suff)	0.166 (0.003)	0.152 (0.012)	**83.74** (0.84)	**94.16** (0.39)
	SLM-FP (Comp+Suff)	0.302 (0.039)	**0.113** (0.013)	82.55 (0.84)	93.68 (0.67)
Suff Criterion	SLM-FP (KL Div)	**0.306** (0.098)	0.131 (0.005)	82.62 (0.88)	93.06 (0.25)
	SLM-FP (MAE)	0.278 (0.058)	0.143 (0.008)	**82.66** (0.61)	**93.78** (0.13)
	SLM-FP (Margin)	0.302 (0.039)	**0.113** (0.013)	82.55 (0.84)	93.68 (0.67)
SLM Ext Head	SLM-FP (Linear)	0.302 (0.039)	**0.113** (0.013)	82.55 (0.84)	**93.68** (0.67)
	SLM-FP (MLP-2048-2)	**0.323** (0.071)	0.144 (0.012)	83.82 (0.77)	93.67 (0.18)
	SLM-FP (MLP-4096-3)	0.295 (0.057)	0.154 (0.027)	**84.53** (0.61)	93.19 (0.79)

Table 2: **UNIREX Ablation Studies on SST.**

Method	Forward Simulation		Subjective Rating
	Accuracy (%)	Confidence (1-4)	Alignment (1-5)
No Rationale	92.00 (3.35)	3.02 (0.39)	-
SGT+P	80.80 (9.73)	2.34 (0.31)	3.64 (0.28)
A2R+P	41.20 (4.71)	**2.83** (0.28)	2.97 (0.12)
UNIREX (AA-FP)	72.00 (7.78)	2.00 (0.31)	3.26 (0.31)
UNIREX (DLM-FP)	**83.60** (5.41)	2.77 (0.28)	**3.96** (0.22)
Gold	81.20 (3.03)	2.88 (0.30)	4.00 (0.20)

Table 3: **Plausibility User Study on SST.**

Method	Faithfulness			Plausibility		Performance	IG	Compute Efficiency
	CSD (″)	Comp (″)	Suff (#)	AUPRC (″)	TF1 (″)	Acc (″)	Conv. Delta (#)	Inference Time (#)
AA (IG-3)	-0.138 (0.040)	0.119 (0.009)	0.258 (0.031)	49.94 (1.77)	50.75 (0.54)	**93.81** (0.55)	8.07 (1.47)	**7.94E-03** (5.30E-05)
AA (IG-5)	-0.141 (0.031)	0.134 (0.015)	0.275 (0.021)	49.38 (1.00)	50.85 (0.70)	**93.81** (0.55)	6.83 (0.92)	1.15E-02 (6.03E-05)
AA (IG-10)	0.011 (0.043)	0.222 (0.015)	0.210 (0.031)	55.87 (0.51)	52.06 (0.33)	**93.81** (0.55)	6.55 (0.48)	2.08E-02 (1.14E-04)
AA (IG-30)	0.056 (0.058)	0.258 (0.020)	0.202 (0.038)	57.23 (1.16)	52.74 (0.43)	**93.81** (0.55)	4.91 (1.82)	5.80E-02 (3.03E-04)
AA (IG-50)	0.066 (0.057)	0.265 (0.017)	0.199 (0.040)	57.70 (1.02)	52.66 (0.37)	**93.81** (0.55)	2.89 (0.34)	9.54E-02 (5.64E-04)
AA (IG-70)	0.072 (0.055)	0.269 (0.016)	0.197 (0.039)	58.11 (1.21)	52.96 (0.32)	**93.81** (0.55)	2.50 (0.25)	1.33E-01 (8.10E-04)
AA (IG-100)	0.074 (0.055)	0.271 (0.016)	0.197 (0.039)	58.25 (1.27)	53.00 (0.42)	**93.81** (0.55)	2.01 (0.13)	1.89E-01 (1.62E-03)
AA (IG-500)	0.082 (0.055)	0.276 (0.016)	0.195 (0.039)	58.61 (1.10)	**53.25** (0.29)	**93.81** (0.55)	0.99 (0.23)	9.38E-01 (5.44E-03)
AA (IG-1000)	0.083 (0.057)	0.278 (0.017)	0.195 (0.040)	**58.64** (1.15)	53.17 (0.39)	**93.81** (0.55)	**0.74** (0.16)	1.88E+00 (1.67E-03)
UNIREX (AA-F)	**0.120** (0.055)	**0.292** (0.051)	**0.171** (0.038)	48.13 (1.14)	50.96 (0.93)	92.97 (0.44)	8.07 (1.47)	**7.94E-03** (5.30E-05)
UNIREX (DLM-FP)	0.151 (0.056)	**0.319** (0.090)	0.167 (0.036)	**85.80** (0.74)	**72.76** (0.19)	**93.81** (0.54)	-	**8.51E-04** (7.82E-07)
UNIREX (SLM-FP)	**0.189** (0.030)	0.302 (0.039)	**0.113** (0.013)	82.55 (0.84)	70.65 (0.44)	93.68 (0.67)	-	8.81E-04 (5.67E-06)

Table 4: **Compute Efficiency Results on SST.**

Instance Ranking	Instance ID	Method	Rationale	F_{task}'s Prediction	Alignment (1-5)
Top Alignment Mean	30	SGT+P	**good actress** .	positive	5.00 (0.00)
		A2R+P	**good actress** .	positive	5.00 (0.00)
		UNIREX (AA-FP)	**good** actress **.**	positive	4.80 (0.45)
		UNIREX (DLM-FP)	**good actress** .	positive	5.00 (0.00)
		Gold	**good actress** .	positive	5.00 (0.00)
	41	SGT+P	a cop story **that understands the medium amazingly well** .	positive	5.00 (0.00)
		A2R+P	**a cop** story that **understands** the **medium amazingly well** .	positive	4.80 (0.45)
		UNIREX (AA-FP)	**a cop story** that **understands** the medium **amazingly** well **.**	positive	4.20 (0.45)
		UNIREX (DLM-FP)	a **cop** story that **understands the medium amazingly well** .	positive	4.80 (0.45)
		Gold	a cop story that **understands the medium amazingly well .**	positive	5.00 (0.00)
	35	SGT+P	chicago is , in many **ways** , an **admirable** achievement .	positive	4.60 (0.55)
		A2R+P	chicago is , in many ways , an **admirable achievement** .	positive	5.00 (0.00)
		UNIREX (AA-FP)	**ch**icago is , in many **ways** , an admirable achievement .	positive	3.20 (0.45)
		UNIREX (DLM-FP)	chicago is , in many ways , an **admirable achievement** .	positive	5.00 (0.00)
		Gold	chicago is , in many ways , an **admirable achievement** .	positive	5.00 (0.00)
Top Alignment Std	9	SGT+P	**bad** beyond belief and ridiculous beyond description **.**	negative	4.60 (0.55)
		A2R+P	**bad** beyond belief and **ridiculous** beyond description .	positive	1.00 (0.00)
		UNIREX (AA-FP)	bad beyond belief **and** ridiculous beyond description **.**	negative	2.60 (0.89)
		UNIREX (DLM-FP)	**bad** beyond belief and **ridiculous** beyond description .	negative	4.80 (0.45)
		Gold	**bad** beyond belief and **ridiculous** beyond description .	negative	4.80 (0.45)
	12	SGT+P	**these** are names to remember , in order to avoid them in the future .	negative	2.40 (0.89)
		A2R+P	these are names to remember , in order to **avoid** them in the future .	positive	1.20 (0.45)
		UNIREX (AA-FP)	these are names to remember , in order to avoid them in the future **.**	negative	2.60 (0.89)
		UNIREX (DLM-FP)	these are names to remember , in order to **avoid** them in the future .	negative	4.80 (0.45)
		Gold	these are names to remember , in order to **avoid** them in the future .	negative	4.80 (0.45)
	19	SGT+P	the title **'s** lameness should clue you **in** on how **bad the movie** is .	negative	4.00 (0.00)
		A2R+P	the title 's **lameness** should **clue you** in on how **bad** the **movie** is .	positive	1.20 (0.45)
		UNIREX (AA-FP)	the title **'s lame**ness **should** clue you **in on** how bad **the** movie is **.**	negative	2.60 (0.89)
		UNIREX (DLM-FP)	the title 's **lameness** should **clue you** in on how **bad the movie** is .	negative	4.80 (0.45)
		Gold	**the title 's lameness** should clue you in on how **bad** the movie is .	negative	4.80 (0.45)

Table 5: **Qualitative Analysis on SST.**

Method	Composite	Faithfulness			Plausibility			Performance	
	NRG (")	NRG (")	Comp (")	Suff (#)	NRG (")	AUPRC (")	TF1 (")	NRG (")	Acc (")
AA (Grad)	0.488	0.337	0.142 (0.010)	0.256 (0.006)	0.192	58.86 (3.65)	27.40 (0.00)	0.935	93.81 (0.55)
AA (Input*Grad)	0.420	0.107	0.078 (0.013)	0.342 (0.014)	0.218	44.16 (1.43)	45.02 (0.39)	0.935	93.81 (0.55)
AA (DeepLIFT)	0.453	0.122	0.085 (0.006)	0.340 (0.018)	0.302	46.50 (1.32)	50.18 (0.32)	0.935	93.81 (0.55)
AA (IG)	0.526	0.297	0.119 (0.009)	0.258 (0.031)	0.347	49.94 (1.77)	50.75 (0.54)	0.935	93.81 (0.55)
L2E	0.557	0.487	0.012 (0.004)	0.009 (0.024)	0.250	44.84 (0.32)	47.24 (0.87)	0.935	93.81 (0.55)
SGT	0.632	0.555	0.147 (0.024)	0.113 (0.031)	0.371	51.38 (2.47)	51.35 (1.64)	0.971	94.40 (0.57)
FRESH	0.330	0.837	0.219 (0.057)	0.000 (0.000)	0.152	42.06 (8.84)	41.19 (4.01)	0.000	78.78 (6.48)
A2R	0.479	0.941	0.283 (0.104)	0.000 (0.000)	0.457	63.36 (6.01)	46.74 (6.65)	0.038	79.39 (11.67)
UNIREX (AA-F)	0.639	0.706	0.292 (0.051)	0.171 (0.038)	0.329	48.13 (1.14)	50.96 (0.93)	0.882	92.97 (0.44)
SGT+P	0.596	0.507	0.139 (0.032)	0.137 (0.026)	0.355	50.38 (1.45)	50.98 (0.46)	0.928	93.70 (0.88)
FRESH+P	0.582	0.765	0.175 (0.043)	0.000 (0.000)	0.970	84.35 (0.87)	71.54 (0.53)	0.011	78.95 (5.18)
A2R+P	0.695	0.953	0.290 (0.016)	0.000 (0.000)	0.978	85.56 (1.01)	70.97 (1.03)	0.154	81.26 (0.52)
UNIREX (DLM-P)	0.770	0.339	0.142 (0.008)	0.255 (0.007)	0.970	84.35 (0.87)	71.54 (0.53)	1.000	94.86 (0.41)
UNIREX (AA-FP)	0.636	0.339	0.296 (0.067)	0.185 (0.048)	0.315	47.60 (2.44)	50.23 (2.26)	0.900	93.25 (0.45)
UNIREX (DLM-FP)	0.897	0.756	0.319 (0.090)	0.167 (0.036)	1.000	85.80 (0.74)	72.76 (0.19)	0.935	93.81 (0.54)
UNIREX (SLM-FP)	0.891	0.807	0.302 (0.039)	0.113 (0.013)	0.940	82.55 (0.84)	70.65 (0.44)	0.927	93.68 (0.67)

Table 6: **Main Results on SST.**

Method	Composite	Faithfulness			Plausibility			Performance	
	NRG (″)	NRG (″)	Comp (″)	Suff (#)	NRG (″)	AUPRC (″)	TF1 (″)	NRG (″)	F1 (″)
AA (Grad)	0.481	0.457	0.184 (0.023)	0.107 (0.017)	0.028	13.31 (0.91)	5.02 (0.00)	0.957	95.33 (0.65)
AA (Input*Grad)	0.503	0.359	0.148 (0.031)	0.137 (0.019)	0.194	8.68 (0.37)	37.58 (0.55)	0.957	95.33 (0.65)
AA (DeepLIFT)	0.468	0.259	0.122 (0.029)	0.172 (0.022)	0.187	9.00 (0.16)	36.15 (1.45)	0.957	95.33 (0.65)
AA (IG)	0.439	0.173	0.134 (0.016)	0.219 (0.044)	0.188	8.88 (0.21)	36.39 (1.29)	0.957	95.33 (0.65)
L2E	0.550	0.445	0.000 (0.007)	0.026 (0.015)	0.248	16.68 (10.20)	38.92 (4.07)	0.957	95.33 (0.65)
SGT	0.553	0.474	0.124 (0.053)	0.071 (0.064)	0.184	10.05 (1.23)	34.64 (1.67)	1.000	96.33 (0.76)
FRESH	0.645	0.732	0.234 (0.034)	0.000 (0.000)	0.305	17.02 (6.22)	48.26 (5.87)	0.899	94.00 (1.44)
A2R	0.431	0.764	0.267 (0.050)	0.000 (0.000)	0.244	35.44 (21.69)	19.78 (25.56)	0.284	79.78 (7.14)
UNIREX (AA-F)	0.601	0.744	0.505 (0.134)	0.122 (0.100)	0.189	9.14 (2.51)	36.28 (1.84)	0.870	93.33 (1.61)
SGT+P	0.586	0.604	0.152 (0.013)	0.022 (0.004)	0.183	9.16 (1.59)	35.33 (0.41)	0.971	95.66 (1.16)
FRESH+P	0.587	0.691	0.193 (0.062)	0.000 (0.000)	1.000	94.32 (0.12)	89.53 (1.63)	0.070	74.84 (12.22)
A2R+P	0.585	0.764	0.267 (0.076)	0.000 (0.000)	0.991	93.53 (0.93)	88.77 (1.22)	0.000	73.22 (0.75)
UNIREX (DLM-P)	0.667	0.024	0.024 (0.003)	0.238 (0.004)	1.000	94.32 (0.12)	89.53 (1.63)	0.978	95.83 (0.29)
UNIREX (AA-FP)	0.543	0.514	0.428 (0.174)	0.195 (0.105)	0.193	8.53 (0.46)	37.71 (3.12)	0.921	94.50 (1.00)
UNIREX (DLM-FP)	0.744	0.326	0.283 (0.217)	0.216 (0.005)	0.991	93.65 (0.36)	88.68 (2.29)	0.913	94.33 (1.61)
UNIREX (SLM-FP)	0.754	0.362	0.313 (0.059)	0.213 (0.014)	0.965	91.70 (1.84)	86.17 (1.20)	0.935	94.83 (0.76)

Table 7: **Main Results on Movies.**

Method	Composite	Faithfulness			Plausibility			Performance	
	NRG (″)	NRG (″)	Comp (″)	Suff (#)	NRG (″)	AUPRC (″)	TF1 (″)	NRG (″)	Acc (″)
AA (Grad)	0.537	0.504	0.331 (0.012)	0.352 (0.007)	0.130	37.33 (0.62)	22.65 (0.00)	0.977	63.56 (1.27)
AA (Input*Grad)	0.573	0.361	0.249 (0.018)	0.385 (0.008)	0.383	39.56 (0.54)	44.43 (0.40)	0.977	63.56 (1.27)
AA (DeepLIFT)	0.605	0.346	0.254 (0.035)	0.403 (0.042)	0.491	42.82 (1.83)	51.72 (1.26)	0.977	63.56 (1.27)
AA (IG)	0.578	0.327	0.216 (0.007)	0.378 (0.010)	0.429	40.07 (5.47)	48.34 (3.16)	0.977	63.56 (1.27)
L2E	0.544	0.493	0.005 (0.003)	0.010 (0.008)	0.161	23.56 (1.09)	37.80 (1.10)	0.977	63.56 (1.27)
SGT	0.618	0.367	0.197 (0.040)	0.324 (0.015)	0.491	43.68 (4.68)	51.00 (3.05)	0.995	64.35 (0.46)
FRESH	0.302	0.546	0.037 (0.036)	0.000 (0.000)	0.261	32.35 (7.66)	39.37 (0.70)	0.101	24.81 (3.46)
A2R	0.277	0.516	0.014 (0.021)	0.000 (0.000)	0.282	41.61 (3.85)	33.12 (9.06)	0.032	21.77 (1.31)
UNIREX (AA-F)	0.690	0.538	0.297 (0.141)	0.286 (0.084)	0.554	46.97 (3.41)	53.99 (1.66)	0.978	63.58 (0.61)
SGT+P	0.601	0.367	0.201 (0.032)	0.328 (0.022)	0.436	41.30 (6.70)	47.95 (1.65)	1.000	64.57 (0.33)
FRESH+P	0.504	0.515	0.013 (0.021)	0.013 (0.021)	0.997	76.07 (1.63)	69.76 (0.27)	0.000	20.36 (0.66)
A2R+P	0.488	0.500	0.001 (0.001)	0.000 (0.000)	0.951	73.59 (0.81)	67.63 (1.54)	0.012	20.91 (0.48)
UNIREX (DLM-P)	0.751	0.267	0.180 (0.016)	0.390 (0.035)	0.997	76.07 (1.63)	69.76 (0.27)	0.990	64.13 (0.46)
UNIREX (AA-FP)	0.685	0.551	0.395 (0.109)	0.381 (0.101)	0.537	45.21 (4.46)	53.91 (3.23)	0.968	63.14 (0.33)
UNIREX (DLM-FP)	0.814	0.492	0.293 (0.043)	0.321 (0.070)	0.997	76.38 (0.57)	69.52 (0.24)	0.953	62.50 (1.34)
UNIREX (SLM-FP)	0.807	0.494	0.390 (0.087)	0.424 (0.110)	0.983	75.12 (0.41)	69.25 (0.41)	0.944	62.09 (2.12)

Table 8: **Main Results on CoS-E.**

Method	Composite	Faithfulness			Plausibility			Performance	
	NRG (″)	NRG (″)	Comp (″)	Suff (#)	NRG (″)	AUPRC (″)	TF1 (″)	NRG (″)	F1 (″)
AA (Grad)	0.498	0.462	0.222 (0.028)	0.120 (0.018)	0.035	22.27 (0.17)	13.81 (0.00)	0.997	69.80 (0.60)
AA (Input*Grad)	0.506	0.289	0.225 (0.048)	0.260 (0.059)	0.231	18.51 (0.23)	43.45 (0.05)	0.997	69.80 (0.60)
AA (DeepLIFT)	0.493	0.249	0.225 (0.012)	0.292 (0.014)	0.234	18.80 (0.19)	43.51 (0.04)	0.997	69.80 (0.60)
AA (IG)	0.499	0.280	0.162 (0.086)	0.222 (0.086)	0.220	18.71 (0.40)	41.79 (1.33)	0.997	69.80 (0.60)
L2E	0.522	0.366	0.007 (0.006)	0.042 (0.024)	0.205	24.48 (2.71)	32.63 (6.12)	0.997	69.80 (0.60)
SGT	0.594	0.564	0.214 (0.105)	0.033 (0.077)	0.224	18.60 (0.42)	42.42 (0.51)	0.995	69.73 (0.13)
FRESH	0.675	0.571	0.176 (0.029)	0.000 (0.000)	0.617	24.68 (7.98)	48.02 (3.04)	0.838	64.47 (3.41)
A2R	0.217	0.404	-0.010 (0.029)	0.000 (0.000)	0.249	18.72 (0.67)	45.45 (0.02)	0.000	36.39 (0.00)
UNIREX (AA-F)	0.711	0.956	0.505 (0.050)	-0.071 (0.020)	0.236	18.82 (0.40)	43.68 (0.38)	0.939	66.17 (4.58)
SGT+P	0.630	0.665	0.280 (0.029)	0.283 (0.039)	0.226	18.63 (0.52)	42.71 (0.39)	1.000	69.91 (0.81)
FRESH+P	0.491	0.413	0.000 (0.013)	0.000 (0.000)	0.999	71.80 (0.27)	77.94 (0.57)	0.060	38.41 (5.34)
A2R+P	0.516	0.422	0.011 (0.024)	0.000 (0.000)	0.977	70.86 (1.30)	76.21 (1.68)	0.150	41.42 (8.73)
UNIREX (DLM-P)	0.708	0.123	0.127 (0.010)	0.322 (0.017)	0.999	71.80 (0.27)	77.94 (0.57)	1.000	69.91 (0.76)
UNIREX (AA-FP)	0.706	1.000	0.545 (0.045)	-0.077 (0.099)	0.231	19.13 (0.71)	42.66 (1.18)	0.888	66.17 (4.58)
UNIREX (DLM-FP)	0.751	0.327	0.135 (0.072)	0.165 (0.029)	0.998	71.89 (0.41)	77.63 (0.62)	0.929	67.53 (1.06)
UNIREX (SLM-FP)	0.784	0.377	0.198 (0.038)	0.171 (0.027)	0.997	71.69 (0.21)	77.79 (0.09)	0.979	69.20 (1.58)

Table 9: **Main Results on MultiRC.**

Method	Composite	Faithfulness			Plausibility			Performance	
	NRG (″)	NRG (″)	Comp (″)	Suff (#)	NRG (″)	AUPRC (″)	TF1 (″)	NRG (″)	F1 (″)
AA (Grad)	0.587	0.518	0.313 (0.009)	0.380 (0.025)	0.244	59.80 (1.32)	15.27 (0.00)	0.999	90.78 (0.27)
AA (Input*Grad)	0.503	0.287	0.205 (0.005)	0.446 (0.020)	0.223	32.98 (1.37)	43.13 (0.86)	0.999	90.78 (0.27)
AA (DeepLIFT)	0.508	0.270	0.195 (0.012)	0.448 (0.014)	0.254	33.47 (1.31)	46.44 (0.04)	0.999	90.78 (0.27)
AA (IG)	0.596	0.473	0.308 (0.011)	0.414 (0.020)	0.317	47.83 (1.04)	37.87 (1.39)	0.999	90.78 (0.27)
L2E	0.606	0.460	0.009 (0.015)	0.036 (0.022)	0.358	58.11 (0.97)	31.35 (0.27)	0.999	90.78 (0.27)
SGT	0.595	0.503	0.288 (0.025)	0.361 (0.038)	0.298	42.46 (3.03)	41.70 (1.78)	0.985	90.23 (0.16)
FRESH	0.518	0.661	0.120 (0.075)	0.000 (0.000)	0.361	38.77 (6.82)	53.71 (3.30)	0.530	72.92 (8.71)
A2R	0.273	0.564	0.053 (0.048)	0.000 (0.000)	0.256	48.48 (11.14)	29.54 (24.72)	0.000	52.72 (14.08)
UNIREX (AA-F)	0.622	0.539	0.330 (0.018)	0.383 (0.055)	0.340	45.29 (3.02)	43.69 (1.98)	0.987	90.31 (0.19)
SGT+P	0.608	0.524	0.286 (0.034)	0.339 (0.032)	0.311	43.03 (1.69)	42.59 (1.63)	0.988	90.36 (0.08)
FRESH+P	0.746	0.695	0.143 (0.072)	0.000 (0.000)	1.000	87.85 (0.13)	77.63 (0.35)	0.544	73.44 (12.88)
A2R+P	0.800	0.751	0.182 (0.097)	0.000 (0.000)	0.992	87.30 (0.44)	77.31 (0.72)	0.656	77.31 (0.72)
UNIREX (DLM-P)	0.842	0.525	0.311 (0.011)	0.371 (0.032)	1.000	87.85 (0.13)	77.63 (0.35)	1.000	90.80 (0.33)
UNIREX (AA-FP)	0.626	0.529	0.341 (0.008)	0.406 (0.046)	0.363	44.79 (0.81)	47.18 (0.83)	0.985	90.21 (0.08)
UNIREX (DLM-FP)	0.857	0.588	0.335 (0.018)	0.346 (0.023)	0.991	86.99 (0.40)	77.53 (0.15)	0.992	90.51 (0.12)
UNIREX (SLM-FP)	0.864	0.603	0.353 (0.017)	0.356 (0.015)	0.994	87.58 (0.14)	77.22 (0.28)	0.994	90.59 (0.09)

Table 10: **Main Results on e-SNLI.**

Ablation	Method	Performance	Faithfulness			Plausibility	
		Acc (″)	CSD (″)	Comp (″)	Suff (#)	AUPRC (″)	TF1 (″)
Ext Type (F)	UNIREX (AA-F, Rand)	94.05 (0.35)	-0.156 (-0.156)	0.171 (0.040)	0.327 (0.050)	44.92 (0.00)	46.15 (0.00)
	UNIREX (AA-F, Gold)	93.81 (0.54)	-0.017 (0.070)	0.232 (0.088)	0.249 (0.021)	100.00 (0.00)	100.00 (0.00)
	UNIREX (AA-F, Inv)	93.47 (1.81)	-0.115 (0.018)	0.242 (0.010)	0.357 (0.019)	20.49 (0.00)	0.00 (0.00)
	UNIREX (AA-F, IG)	93.81 (0.55)	-0.138 (0.040)	0.119 (0.009)	0.258 (0.031)	49.94 (1.77)	50.75 (0.54)
Ext Type (FP)	UNIREX (AA-FP, Sum)	93.81 (0.55)	-0.138 (0.040)	0.119 (0.009)	0.258 (0.031)	49.94 (1.77)	50.75 (0.54)
	UNIREX (AA-FP, MLP)	93.23 (0.92)	0.087 (0.134)	0.285 (0.051)	0.197 (0.100)	54.82 (1.97)	49.62 (0.65)
	UNIREX (DLM-FP)	93.81 (0.18)	0.151 (0.056)	0.319 (0.090)	0.167 (0.036)	85.80 (0.74)	72.76 (0.19)
	UNIREX (SLM-FP)	93.68 (0.67)	0.189 (0.030)	0.302 (0.039)	0.113 (0.013)	82.55 (0.84)	70.65 (0.44)
Comp/Suff Loss	UNIREX (SLM-FP, Comp)	93.59 (0.11)	0.040 (0.096)	0.350 (0.048)	0.310 (0.049)	82.79 (0.62)	70.74 (0.81)
	UNIREX (SLM-FP, Suff)	94.16 (0.39)	0.014 (0.010)	0.166 (0.003)	0.152 (0.012)	83.74 (0.84)	70.94 (0.86)
	UNIREX (SLM-FP, Comp+Suff)	93.68 (0.67)	0.189 (0.030)	0.302 (0.039)	0.113 (0.013)	82.55 (0.84)	70.65 (0.44)
Suff Criterion	UNIREX (SLM-FP, KL Div)	93.06 (0.25)	0.174 (0.100)	0.306 (0.098)	0.131 (0.005)	82.62 (0.88)	70.43 (0.65)
	UNIREX (SLM-FP, MAE)	93.78 (0.13)	0.135 (0.053)	0.278 (0.058)	0.143 (0.008)	82.66 (0.61)	70.25 (0.45)
	UNIREX (SLM-FP, Margin)	93.68 (0.67)	0.189 (0.030)	0.302 (0.039)	0.113 (0.013)	82.55 (0.84)	70.65 (0.44)
SLM Ext Head	UNIREX (SLM-FP, Linear)	93.68 (0.67)	0.189 (0.030)	0.302 (0.039)	0.113 (0.013)	82.55 (0.84)	70.65 (0.44)
	UNIREX (SLM-FP, MLP-2048-2)	93.67 (0.18)	0.179 (0.060)	0.323 (0.071)	0.144 (0.012)	83.82 (0.77)	70.93 (0.87)
	UNIREX (SLM-FP, MLP-4096-3)	93.19 (0.79)	0.141 (0.030)	0.295 (0.057)	0.154 (0.027)	84.53 (0.61)	71.41 (0.91)

Table 11: **UNIREX Ablation Studies on SST (Full).**

Task	Dataset	Method	Performance	Faithfulness		
			Perf (″)	CSD (″)	Comp (″)	Suff (#)
Sentiment Analysis	SST	Vanilla	93.81 (0.74)	-0.070 (0.061)	0.145 (0.023)	0.215 (0.038)
		UNIREX (AA-F)	93.19 (0.40)	0.360 (0.055)	0.405 (0.031)	0.045 (0.024)
		UNIREX (DLM-FP)	93.81 (0.18)	0.151 (0.056)	0.319 (0.090)	0.167 (0.036)
		UNIREX (SLM-FP)	93.68 (0.67)	0.189 (0.030)	0.302 (0.039)	0.113 (0.013)
	Yelp	Vanilla	92.50 (2.07)	-0.156 (0.028)	0.067 (0.004)	0.222 (0.031)
		UNIREX (AA-F)	90.75 (1.30)	-0.138 (0.120)	0.096 (0.026)	0.233 (0.096)
		UNIREX (DLM-FP)	92.37 (0.46)	0.169 (0.060)	0.265 (0.094)	0.097 (0.033)
		UNIREX (SLM-FP)	86.60 (1.57)	0.114 (0.056)	0.175 (0.055)	0.060 (0.001)
	Amazon	Vanilla	91.13 (0.28)	-0.120 (0.038)	0.096 (0.008)	0.217 (0.033)
		UNIREX (AA-F)	86.60 (0.95)	-0.111 (0.161)	0.100 (0.042)	0.210 (0.122)
		UNIREX (DLM-FP)	89.35 (2.22)	0.133 (0.039)	0.232 (0.072)	0.098 (0.033)
		UNIREX (SLM-FP)	81.82 (7.62)	0.097 (0.027)	0.147 (0.012)	0.050 (0.017)
Hate Speech Detection	Stormfront	Vanilla	10.48 (1.66)	-0.066 (0.072)	0.153 (0.002)	0.219 (0.071)
		UNIREX (AA-F)	9.43 (1.45)	0.329 (0.104)	0.337 (0.073)	0.008 (0.031)
		UNIREX (DLM-FP)	10.37 (2.66)	0.052 (0.027)	0.167 (0.084)	0.115 (0.059)
		UNIREX (SLM-FP)	4.51 (1.87)	0.049 (0.041)	0.110 (0.039)	0.062 (0.043)
Offensive Speech Detection	OffenseEval	Vanilla	33.51 (0.99)	-0.125 (0.068)	0.104 (0.007)	0.229 (0.064)
		UNIREX (AA-F)	35.69 (2.30)	-0.028 (0.084)	0.076 (0.008)	0.104 (0.076)
		UNIREX (DLM-FP)	35.52 (1.26)	0.053 (0.012)	0.140 (0.049)	0.087 (0.045)
		UNIREX (SLM-FP)	38.17 (0.96)	0.039 (0.031)	0.087 (0.016)	0.048 (0.024)
Irony Detection	SemEval2018-Irony	Vanilla	29.63 (4.72)	-0.058 (0.075)	0.154 (0.001)	0.212 (0.074)
		UNIREX (AA-F)	47.99 (6.33)	0.026 (0.080)	0.087 (0.022)	0.061 (0.071)
		UNIREX (DLM-FP)	31.97 (2.80)	0.047 (0.017)	0.149 (0.052)	0.102 (0.053)
		UNIREX (SLM-FP)	17.42 (4.04)	0.027 (0.047)	0.091 (0.027)	0.064 (0.033)

Table 12: **Zero-Shot Explainability Transfer from SST to Unseen Datasets/Tasks (Full).**