KACE: Generating Knowledge-Aware Contrastive Explanations for Natural Language Inference

Qianglong Chen[1,2*] **Feng Ji**[3†] **Xiangji Zeng**[1], **Feng-Lin Li**[2],
Ji Zhang[2], **Haiqing Chen**[2], **Yin Zhang**[1‡]
[1]College of Computer Science and Technology, Zhejiang University, China
[2]DAMO Academy, Alibaba Group, China, [3]Tencent, China
{chenqianglong,zengxiangji,zhangyin98}@zju.edu.cn
{fenglin.lfl,zj122146,haiqing.chenhq}@alibaba-inc.com
{neilji}@tencent.com

Abstract

In order to better understand the reason behind model behaviors (i.e., making predictions), most recent work has exploited generative models to provide complementary explanations. However, existing approaches in natural language processing (NLP) mainly focus on "WHY A" rather than contrastive "WHY A NOT B", which is shown to be able to better distinguish confusing candidates and improve model performance in other research fields. In this paper, we focus on generating **C**ontrastive **E**xplanations with counterfactual examples in NLI and propose a novel **K**nowledge-**A**ware generation framework (**KACE**). Specifically, we first identify rationales (i.e., key phrases) from input sentences, and use them as key perturbations for generating counterfactual examples. After obtaining qualified counterfactual examples, we take them along with original examples and external knowledge as input, and employ a knowledge-aware generative pre-trained language model to generate contrastive explanations. Experimental results show that contrastive explanations are beneficial to clarify the difference between predicted answer and other answer options. Moreover, we train an BERT-large based NLI model enhanced with contrastive explanations and achieve an accuracy of 91.9% on SNLI, gaining an improvement of 5.7% against ETPA ("Explain-Then-Predict-Attention") and 0.6% against NILE ("WHY A").

1 Introduction

In recent years, pre-trained language models (Devlin et al., 2019; Liu et al., 2019; Yang et al., 2019) have been widely adopted in many tasks of natural language processing (Talmor et al., 2019; Choi et al., 2018; Bowman et al., 2015). However, due to the lack of textual explanations, most downstream models become more complicated and difficult to understand. End users, especially those working in critical domains such as healthcare or online education, become more skeptical and reluctant to adopt or trust them, although these models have been proved to improve the decision-making performance. Therefore, providing faithful textual explanations has become a promising way to overcome the black-box property of neural networks, which has attracted the attention of academia and industrial communities.

Recently, the majority of existing methods (Xu et al., 2020; Cheng et al., 2020; Karimi et al., 2020; Ramamurthy et al., 2020; Atanasova et al., 2020; Kumar and Talukdar, 2020) in natural language processing try to explain the predictions of neural models in a model-intrinsic or model-agnostic (also known as post-hoc) way. While post-hoc models (Chen et al., 2020b; Karimi et al., 2020; Kumar and Talukdar, 2020) provide explanations after making predictions without affecting the overall accuracy, most of them neglect the rationales in inputs and provide textual explanations just in the form of "WHY A". However, we argue that contrastive explanations in the form of "WHY A NOT B" could provide more informative and important clues that are easier to understand and persuade end-users. Moreover, we believe that contrastive explanations could benefit downstream tasks (e.g., NLI), since such kind of explanations contain more helpful information (e.g. relations between rationales) that can be used to improve model performance.

To further enhance the explainability and performance of NLI, we propose a novel textual contrastive explanation generation framework in this paper, which is post-hoc and considers rationales, counterfactual examples, and external knowledge. Specifically, we first identify rationales (i.e., key phrases) from a premise-hypothesis (P-H) pair with

[*] Work is done during internship at Alibaba Group.

[†] The work is mainly conducted while being at Alibaba Group.

[‡] Corresponding Author: Yin Zhang

*Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics
and the 11th International Joint Conference on Natural Language Processing*, pages 2516–2527
August 1–6, 2021. ©2021 Association for Computational Linguistics

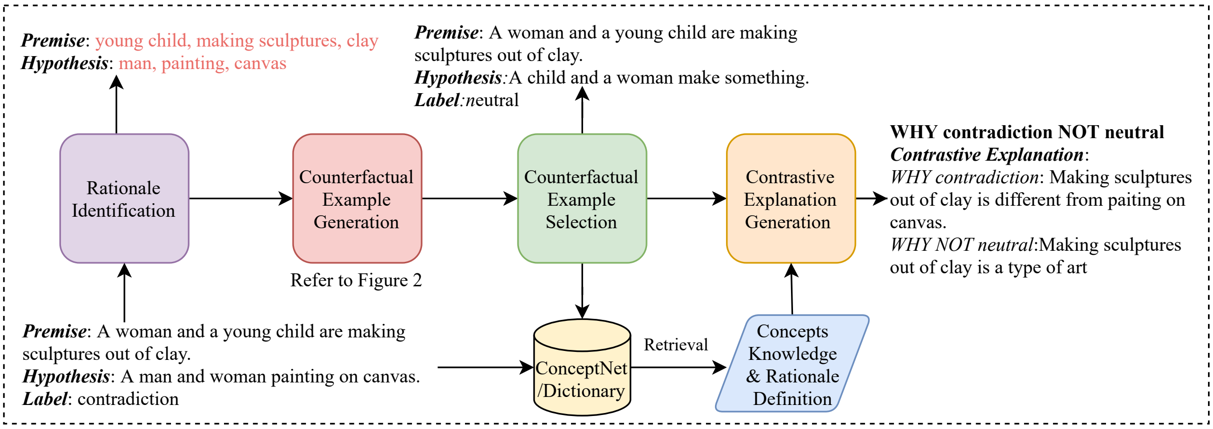
Premise: young child, making sculptures, clay
Hypothesis: man, painting, canvas
Premise: A woman and a young child are making sculptures out of clay.
Hypothesis:A child and a woman make something.
Label:neutral
Rationale Identification
Counterfactual Example Generation
Refer to Figure 2
Counterfactual Example Selection
Contrastive Explanation Generation
WHY contradiction NOT neutral
Contrastive Explanation:
WHY contradiction: Making sculptures out of clay is different from paiting on canvas.
WHY NOT neutral:Making sculptures out of clay is a type of art
Premise: A woman and a young child are making sculptures out of clay.
Hypothesis: A man and woman painting on canvas.
Label: contradiction
ConceptNet /Dictionary
Retrieval
Concepts Knowledge & Rationale Definition

Figure 1: The overall workflow of contrastive explanation generation, which contains rationale identification, counterfactual example generation (as described in Figure 2) and selection, and knowledge-aware contrastive explanation generation. In our "WHY A NOT B" paradigm, we will generate explanations for A and each other class B (i.e., we will generate "WHY NOT neutral" and "WHY NOT entailment" in this example). The counterfactual example selection aims to select one most qualified for any other class B.

label A, and then use them as the key perturbations for transforming and generating candidate counterfactual examples. Then we further select one most qualified counterfactual example for any other label B. Note that the acquisition of a qualified counterfactual example of class B is essential to generate a meaningful explanation for "WHY NOT B", otherwise the resultant contrastive explanation will be groundless or useless. After that, we take the selected examples along with the original P-H pair and related external knowledge as input, and finally employ a knowledge-aware pre-trained language model to generate contrastive explanation, which will specify why the prediction label is A rather than B, and clarify the confusions for end-users. Moreover, we train an NLI model enhanced with contrastive explanations and achieve the new state-of-art performance on SNLI.

The contributions of this paper are as follows:

- We introduce a novel knowledge-aware contrastive explanation generation framework (**KACE**) for natural language inference tasks.

- We consider the rationales in inputs and regard them as important perturbations for generating counterfactual examples rather than just discarding them like previous post-hoc work (Hendricks et al., 2018; Cheng et al., 2020).

- We integrate external knowledge with generative pre-trained language model rather than only taking original inputs (Kumar and Talukdar, 2020; Rajani et al., 2019) for contrastive explanation generation.

- Experimental results show that knowledge-aware contrastive explanations are able to clarify the difference between predicted class and the others, which help to clarify the confusion of end-users and further improve model performance than "WHY A" explanations[1].

2 Task Definition and Overall Workflow

Here, we define the task of contrastive explanation generation for NLI. Given a trained neural network model f with input x and predicted class A, the problem of generating contrastive explanations (CE) to an input x is to specify why x belongs to category/class A rather than B, defined as:

$$r = Rationales(x, A) \tag{1}$$

$$x' = Reversal(x, B, r) \tag{2}$$

$$CE = Generator(x', x, A) \tag{3}$$

In Equation 1, we first identify a set of rationales in given inputs, as described in Section 3.1, and in Equation 2 we generate counterfactual examples with reversal mechanism as presented in Section 3.2. In Equation 3, we take the selected counterfactual example along with original example and external knowledge as input, and employ a knowledge-aware generator to produce contrastive explanation as detailed in Section 3.3.

[1] Our code will be released as soon as possible at https://github.com/AI4NLP/KACE

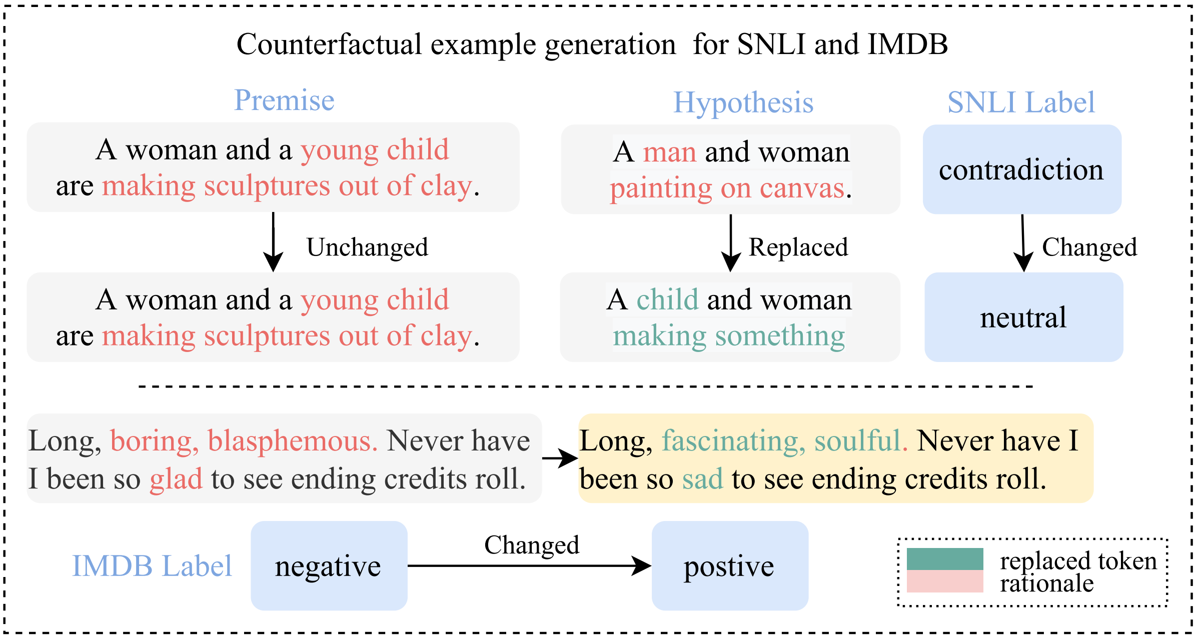
Counterfactual example generation for SNLI and IMDB
Premise
Hypothesis
SNLI Label
A woman and a young child are making sculptures out of clay.
A man and woman painting on canvas.
contradiction
Unchanged
Replaced
Changed
A woman and a young child are making sculptures out of clay.
A child and woman making something
neutral
Long, boring, blasphemous. Never have I been so glad to see ending credits roll.
Long, fascinating, soulful. Never have I been so sad to see ending credits roll.
IMDB Label
negative
Changed
postive
replaced token
rationale

Figure 2: Counterfactual example generation for SNLI and IMDB.

3 Approach

3.1 Rationale Identification

Considering that rationales are important features of an instance, it is essential to regard rationales as key perturbations for counterfactual example generation. In this paper, we formulate rationale identification as a token-level sequence labelling task where 1 indicates a rationale token and 0 indicates a background token.

Being similar with (Thorne et al., 2019), we first construct the input sequence for a premise p and a hypothesis h as S^p=$\langle s\rangle$ $Label$ $\langle s\rangle$ $Premise$ $\langle s\rangle$ and S^h=$\langle s\rangle$ $Hypothesis$ $\langle s\rangle$, where $\langle s\rangle$ is a special token that separates the components. Let y represent the relation between S^p and S^h where $y \in$ {entailment, contradiction, neutral}. For each instance, we need to identify a subset r of zero or more tokens as rationales from both premise and hypothesis sentences. Both premise and hypothesis are encoded with RoBERTa (Liu et al., 2019), yielding hidden representation $\mathbf{H}^p$=$[\cdots, h_j^p, \cdots]$ and $\mathbf{H}^h$=$[\cdots, h_i^h, \cdots]$ respectively.

As rationalizer is proposed by (Zhao and Vydiswaran, 2021), we follow this work for rationale identification using cross attention to embed the hypothesis (premise) into premise (hypothesis), which is defined as:

$$a_{ij} = \frac{exp((h_i^h)^T Tanh(W_1^T h_j^p))}{\sum_{m=0}^{L^p} exp((h_i^h)^T Tanh(W_1^T h_m^p))} \tag{4}$$

$$\hat{h}_i^h = [h_i^h, Pooling(\mathbf{H}^p), \sum_k a_{ij} h_j^p] \tag{5}$$

where a_{ij} denotes the attention score of j^{th} token in premise to the i^{th} token in the hypothesis, L^p denotes the length of the premise sentence and W_1 is a trainable parameter matrix. The representation of i^{th} token in the hypothesis, denoted as $\hat{h}_i^h$, is created by concatenating its original state representation, max-pooling representation over h^p, and the corresponding sum of attention representation from h^p. At last, we use a softmax layer with a linear transformation to model the probability of the i^{th} token in S^h being a rationale token.

3.2 Counterfactual Example Generation

As we have introduced above, counterfactual examples of other classes are of key importance to generate contrastive explanations. In this part, we describe how to generate counterfactual examples.

Given a trained neural network model f, the problem of generating counterfactual example for an instance x is to find a set of examples $c_1, c_2, ..., c_k$ that lead to a desired prediction y'. The counterfactual examples are explainable and contrastive when they appropriately consider proximity, diversity and validity.

Here, we define a three-part loss function to select qualified counterfactual example:

$$L = L_{valid} + \lambda_1 L_{dist} + \lambda_2 L_{div} \tag{6}$$

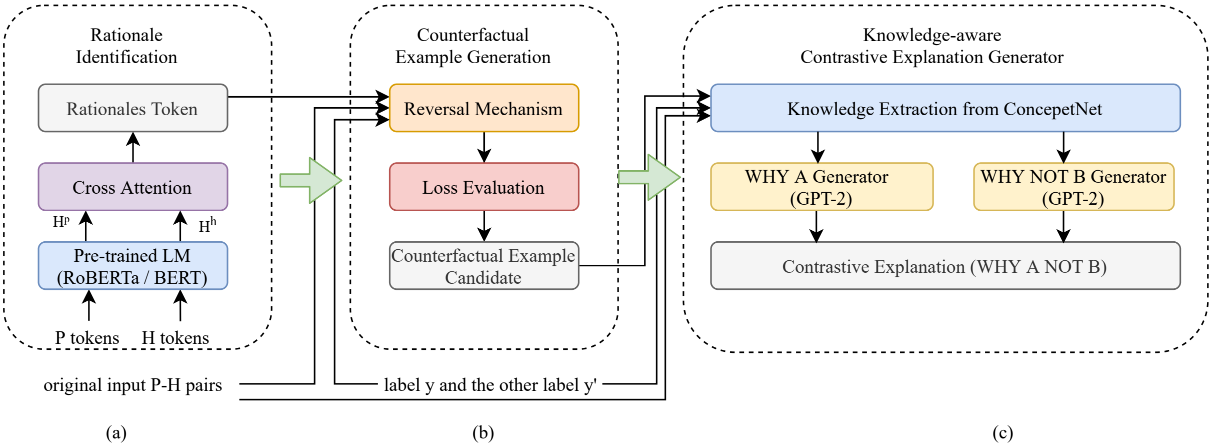
Rationale Identification
Rationales Token
Cross Attention
H^p
H^h
Pre-trained LM (RoBERTa / BERT)
P tokens
H tokens
Counterfactual Example Generation
Reversal Mechanism
Loss Evaluation
Counterfactual Example Candidate
Knowledge-aware Contrastive Explanation Generator
Knowledge Extraction from ConcepetNet
WHY A Generator (GPT-2)
WHY NOT B Generator (GPT-2)
Contrastive Explanation (WHY A NOT B)
original input P-H pairs
label y and the other label y'
(a)
(b)
(c)

Figure 3: Details of the approach. The framework consists of: a) rationale identification, b) counterfactual example generation and selection, c) knowledge-aware contrastive explanation generation. Given the original input P-H pairs and the annotated label A, an other label B, the approach identify the rationales, generate counterfactual examples and produce contrastive explanation based on them.

where λ_1 and λ_2 are hyperparameters for balancing L_{dist} and L_{div}. For generating counterfactual example, the validity term, which ensures the generated counterfactual examples have desired prediction target, is defined as:

$$L_{valid} = \sum_{i=1}^{k} loss(f(c_i), y') \tag{7}$$

Meanwhile, the generated examples should be proximal to the original instance as described in (Cheng et al., 2020), which means only a small change needs to be made. We do not expect a big change that transforms a large portion of the original, in which way there will be no difference with merely presenting an example of counter classes and the corresponding explanation will be uninformative or useless. That is, we expect that resultant examples are able to preserve the main content of input while changing domain-related parts.

$$L_{dist} = \sum_{i=1}^{k} dist(c_i, x) \tag{8}$$

In this paper, we choose a weighted Heterogeneous Manhattan-Overlay Metric (Wilson and Martinez, 1997) to calculate the distance as follows:

$$dist(c, x) = \sum_{t} d_t(c^t, x^t) \tag{9}$$

where t indicates a rationale.

To achieve diversity, we want generated examples to be different from each other. Specifically, we calculate the pairwise distance of a set of counterfactual examples and minimize:

$$L_{div} = -\frac{1}{k}\sum_{i=1}^{k}\sum_{j=i}^{k} dist(c_i, c_j) \tag{10}$$

After defining the loss function, we use a reversal mechanism to produce counterfactual examples. In the reversal mechanism, we use hypernym and hyponym of tokens in WordNet[2] for perturbation.

For example, as shown in Figure 2, the original premise and hypothesis are “a woman and a young child are making sculptures out of clay” and “a man and a woman painting on canvas”, and the label is “contradiction”. We find from WordNet the hypernyms of “making sculptures out of clay” and “painting on canvas” as “doing art” and “making something” respectively. We replace them with their hypernyms to obtain counterfactual examples, and use the model f trained on the original P-H training dataset to predict the resultant examples (Equation 7), and keep those belong to neutral or entailment. After the validity justification, we perform further selection by following Equation 8 and Equation 10, and choose the samples with the smallest loss for neutral and entailment for latter contrastive explanation generation.

3.3 Contrastive Explanation Generation

After obtaining qualified counterfactual examples, some work (Cheng et al., 2020; Wachter et al.,

[2] https://wordnet.princeton.edu/

2017; Verma et al., 2020) provides them as counterfactual explanation directly. However, since counterfactual examples do not provide explanations explicitly, it could be difficult for users to understand. Hence, in this part, we focus on generating contrastive explanation via knowledge-aware generative language model, which explain "WHY A NOT B" rather than merely "WHY A".

While traditional approach generate explanation with SHAP[3] or LIME[4], recent work has exploited to use pre-trained generative language models (Radford et al., 2019; Lewis et al., 2020; Raffel et al., 2020). In this paper, we use knowledge-aware pre-trained language model to generate contrastive explanation.

Knowledge Extraction Given selected counterfactual examples and identified rationales, we extract relevant knowledge to enhance the generative language model. We acquire structured knowledge and rationale definitions from ConceptNet[5] and dictionary source[6] separately. For ConceptNet, we extract knowledge with Breadth-First-Search (BFS) algorithm as described in (Ji et al., 2020). For dictionary, we extract the definition of rationales by following (Chen et al., 2020a). After extraction, we concatenate these knowledge for training knowledge-aware explanation generator.

Knowledge-Aware Explanation Generator For contrastive explanation generation, we divide the "WHY A NOT B" problem into two simple question: 1) why the label of the input belong to A, 2) why the label of the input not belong to B.

In previous study, (Kumar and Talukdar, 2020) proposed a label-specific explanation generator, which fine-tuned GPT2 independently for each label. However, the generator can only produce explanations for "WHY A". For the other part of contrastive explanation, we collect some contrastive explanations annotated by human and use them to fine-tune a "WHY NOT B" generator.

Taking a premise-hypothesis pair x along with the qualified counterfactual example x' and extracted knowledge K_E as input, which is in the form of $\langle s\rangle\, Label\, \langle s\rangle\, x\, \langle s\rangle\, x'\, \langle s\rangle\, K_E\, \langle s\rangle$, our fine-tuned language model generates explanations that support the corresponding label in a "WHY A NOT B" way. With these explanations, end-users can observe and understand the difference between original input and counterfactual example explicitly.

4 Experiments

4.1 Datasets

4.1.1 Natural Language Inference

SNLI & e-SNLI The SNLI dataset (Bowman et al., 2015) is a balanced collection of P-H annotated pairs with labels from {entailment, neutral, contradiction}, which consists of about 550K, 10K and 10K examples for train, development, and test set, respectively [7]. (Camburu et al., 2018) extend the SNLI dataset to e-SNLI [8] with natural language explanations of the ground truth labels. Annotators were asked to highlight words in the premise and hypothesis pairs which could explain the labels and write a natural language explanation using the highlighted words. In this paper, we use the highlighted words for rationale identification and use the natural language explanation to fine-tune the language model based "WHY A" generator.

IMDB The IMDB dataset (Maas et al., 2011) is a movie reviews dataset for sentiment classification. It contains 25,000 training data and 25,000 test data with movie reviews labeled as positive or negative. In this paper, we use IMDB as a out-of-domain dataset to evaluate if counterfactual examples can improve the robustness of our model.

4.2 Evaluation

We are committed to generate contrastive explanations which can distinguish the predicted label and others at semantic level, hence, BLEU (Papineni et al., 2002) score is not a proper way to measure the quality of explanations. That is, it can be better confirmed by manual evaluation. In this work, we use manual evaluation and case study for contrastive explanations quality evaluation. Meanwhile, we use accuracy to measure the effectiveness of generated contrastive explanations on improving model performance in terms of data augmentation (organized in the form of $\langle s\rangle\, CE\, \langle s\rangle\, Premise\, \langle s\rangle\, Hypothesis\, \langle s\rangle$).

[3] https://github.com/slundberg/shap

[4] https://github.com/marcotcr/lime

[5] https://github.com/commonsense/conceptnet5/

[6] https://dictionary.cambridge.org/

[7] https://nlp.stanford.edu/projects/snli/snli_1.0.zip

[8] https://github.com/OanaMariaCamburu/e-SNLI

Table 1: Different types of explanations, including token-level explanation, e-SNLI explanation and contrastive explanation. The explanation of e-SNLI explains why the label of a given pair is contradiction, while the contrastive explanation specifies why the label is contradiction and not neutral or entailment.

Premise-Hypothesis Pair	A woman and a young child are making sculptures out of clay.(**P**) A man and woman painting on canvas.(**H**)
Label	**contradiction**
	Results of our Approach
Token-level Explanation (Rationales)	**young child, making sculptures, clay, man painting, canvas**
Knowledge from ConceptNet	sculpture is a type of art. canvas is used for art / painting on clay is used for making sculpture.
Counterfactual Examples	A woman and a young child are making sculptures out of clay. (**P**) A child and a woman make something. (perturbed **H**)
Contrastive Explanation (WHY contradiction NOT neutral?)	Making sculptures out of clay different from painting on canvas, Making sculptures out of clay is a type of art.
	Explanation of Other Methods
NILE:post-hoc (WHY contradiction?)	Women are not men.
LIREx-base (WHY contradiction?)	A young child is not a man.
e-SNLI Explanation (Golden Annotated)	A young child is not a man. Making sculptures out of clay is a different type of art and medium than painting on canvas.

4.3 Baselines

4.3.1 Pre-trained Language Model

RoBERTa & BERT For sequence labelling during rationale identification, we use RoBERTa-large and BERT-large, which have 24 layers, 16 attention heads and a hidden size of 1024 (355M parameters for RoBERTa-large, 340M parameters for BERT-large). For downstream classifications tasks, a classification layer is added over the hidden state of the first [CLS] token at the last layer.

GPT-2 For natural language explanation generation, we use the GPT-2 architecture (Radford et al., 2019). In particular, we use the GPT2-medium model that has 24 layers, 16 attention heads and a hidden size of 1024 (345M parameters). We fine-tuned GPT-2 model with label-specific examples that are integrated with contrastive examples and external knowledge from ConceptNet.

4.3.2 NLI Baselines

ESIM & SemBERT & CA-MTL ESIM (Chen et al., 2017) proposes a enhanced sequential inference model that considers recursive architectures in both local inference modeling and inference composition, and incorporates syntactic parsing information. (Zhang et al., 2020) incorporate explicit contextual semantics from pre-trained semantic role labeling and introduce an improved language representation model, Semantics-aware BERT (SemBERT), which is capable of explicitly absorbing contextual semantics with a BERT backbone. CA-MTL (Pilault et al., 2021) is a novel transformer based architecture that consists of a new conditional attention mechanism as well as a set of task conditioned modules that facilitate weight sharing, and achieves the new state-of-art performance on SNLI.

4.4 NLI with Explanation Baselines

ETPA (Camburu et al., 2018) propose Explain-Then-Predict-Attention (ETPA) that generates an explanation and then predicts the label with only the generated explanation.

NILE:post-hoc (Kumar and Talukdar, 2020) propose natural language inference over label-specific explanations (NILE). A premise and hypothesis pair is input to label-specific a candidate explanation generator that generates natural language explanations supporting the corresponding label. The generated explanations are then fed into an explanation processor, which predicts labels using evidence presented in these explanations.

LIREx-base (Zhao and Vydiswaran, 2021) propose LIREx-base that incorporates both a rationale enabled explanation generator and an instance selector to select only relevant, plausible natural language explanations (NLEs) to augment NLI models and evaluate on the standardized SNLI.

4.5 Experiment Setting

For rationale identification, we use RoBERTa-base to extract hidden representations and set the learning rate to 2e-5, dropout to 0.02, batch size to 8 and number of epochs to 10. Meanwhile, we use AdamW (Loshchilov and Hutter, 2018) as the optimizer and adopt cross-entropy loss as the loss function. In the counterfactual example generation part, we build a hypernym and hyponym table, and use hypernym and hyponym of tokens in WordNet for perturbation. In the contrastive explanation generation part, we use GPT-2 as the generative language model for training "WHY A" generator and "WHY NOT B" Generator. For generator, we set the learning rate to 5e-5, adam epsilon to 1e-8, length for generation to 100.

4.6 Results And Analysis

Table 2: Human evaluation of contrastive and baseline explanations on 100 SNLI test samples. Average score of two annotators (%).

Model	**Explanations Quality**
NILE:post-hoc	81.5
LIREx-base	88.5
Contrastive Exp	90.5

Explanation Generation for SNLI In Table 1, we present the inputs of our model, the results of our approach that include token-level explanation (rationales), counterfactual example and generated contrastive explanation, compared with manually annotated explanation and generated "WHY A" explanations by NILE:post-hoc and LIREx-base.

Compared with "WHY A" explanations that are simple and lack essential information, the contrastive explanation contains more information such as "making sculptures out of clay is a type of art" and "making sculptures is different from painting on canvas". As shown in Table 1, we provide not only the contrastive explanation but also the identified rationales and reversed counterfactual example for reference.

To quantitatively assess contrastive explanations, we compared our method with LIREx-base and NILE:post-hoc in terms of explanation quality through human evaluation on 100 SNLI test samples. The explanation quality refers to whether an explanation provides enough essential information for a predicted label. As shown in Table 2, contrastive explanations produced by our method have a better quality by obtaining over 2.0% and 9.0% than LIREx-base and NILE:post-hoc .

Table 3: The accuracy (%) of our method compared with RoBERTa-large and BERT-large on SNLI.

Model	**Dev**	**Test**
Traditional Baseline		
ESIM	88.4	88.6
BERT-large	91.3	91.1
SemBERT-large	92.0	91.6
BERT-wwm	92.1	91.6
SemBERT-wwm	92.2	91.9
CA-MTL	92.4	92.1
WHY A Exp Generator		
ETPA	87.0	86.2
NILE: post-hoc	91.9	91.5
LIREx-base	92.2	91.6
WHY A NOT B Exp Generator		
BERT-large+Contrastive Exp	91.5	**91.9**
RoBERTa-large+Contrastive Exp	92.2	**92.1**
Human Exp Performance		
BERT-large + human Exp	91.6	**92.2**
RoBERTa-large + human Exp	**92.7**	**92.6**

Explanation Enhanced NLI In Table 3, we report the experimental results of our method and other baselines include BERT, SemBERT (Zhang et al., 2020), CA-MTL (Pilault et al., 2021), NILE:post-hoc (Kumar and Talukdar, 2020) and LIREx-base (Zhao and Vydiswaran, 2021) on SNLI. With contrastive explanations, we are able to improve the performance of both BERT-large and RoBERTa-large. Compared with NILE:post-hoc (Kumar and Talukdar, 2020), the same scale

BERT-large model with contrastive explanations brings a gain of 0.4% on test, which indicates the knowledge-aware contrastive generator are better than the generator of NILE. Compared with LIREx-base that uses RoBERTa-large (Zhao and Vydiswaran, 2021), the BERT-large model and RoBERTa-large with contrastive explanations bring a gain of 0.3% and 1.0% separately, which suggests contrastive explanations are better than rationale enabled explanation. In general, contrastive explanations can achieve new state-of-art performance and get it closer to human annotation (a gain of 1.1% on BERT-Large). We believe that contrastive explanations contain more helpful information (e.g., relations between rationales, differences between original and counterfactual examples) that can be used to improve model performance.

Ablation Study We perform ablation studies with BERT-large on the SNLI dataset to evaluate the impacts of different components employed in our method, and report the results in Table 4. We isolated rationales, counterfactual examples and external knowledge, separately. The model without rationales means we generate contrastive explanations with counterfactual examples generated through randomly replacing tokens and extracted external knowledge. The model without counterfactual examples means we extracted knowledge with given rationales and generate contrastive explanation with them. The model without external knowledge means we generate contrastive explanation only with rationales and counterfactual examples. The model without contrastive explanation actually is the BERT-large baseline in SNLI. We can observe that each component is helpful. Especially, if we remove external knowledge and contrastive explanations, we can see a clear decrease of 0.6% and 0.8%, respectively. It indicates that external knowledge and contrastive explanation generation are the most essential components, while rationales and counterfactual examples affect the performance less. On one hand, the ablation study results show, external knowledge and rationales affect more than counterfactual examples on explanation generation. On the other hand, the results suggest that each component contributes positively, and indicate the importance of knowledge aware contrastive explanations, as we highlighted in the title.

Table 4: The accuracy (%) of ablation studies on SNLI.

Model	**Dev**	**Test**
Our Model	91.5	91.9
w/o Rationales	91.0	91.5
w/o Counterfactual Example	91.4	91.6
w/o External Knowledge	91.2	91.3
w/o Contrastive Exp	91.3	91.1

Out of Domain Counterfactual Example In this part, we use the generated counterfactual examples of IMDB for out of domain evaluation. As shown in Table 5, we train BERT-base on two different training sets: the original training set TRAIN$_O$, and the union of original training examples and generated counterfactual examples TRAIN$_{O \cup C}$, and evaluate it with two separated dev sets: the original dev set DEV$_O$ and the generated counterfactual example dev set DEV$_C$. Experimental results shown that BERT-base model enhanced with counterfactual examples achieves 88.5% and 95.1%, bringing a gain of 11.0% on DEV$_C$ while a slight decrease of 1.7% on DEV$_O$. It indicates that counterfactual examples can help to improve the robustness of model for more diversified data distribution.

With IMDB evaluation, we demonstrate that counterfactual examples can not only help to generate contrastive explanation, but also contribute to data augmentation. In the experiments on SNLI, we evaluated the effectiveness of counterfactual example in contrastive explanation generation. In IMDB experiments, we further verify the effectiveness of counter-factual examples for data augmentation with only rationales identification and heuristic reversal mechanism.

Table 5: The accuracy of BERT-base on IMDB, being trained with TRAIN$_O$ and TRAIN$_{O \cup C}$, evaluated on DEV$_O$ and DEV$_C$.

Model	**DEV$_O$**	**DEV$_C$**
BERT-base (TRAIN$_O$)	90.2	86.1
BERT-base (TRAIN$_{O \cup C}$)	88.5	95.1

5 Related Work

5.1 Counterfactual Example Generation

Counterfactual example aims to find a minimal change in data that "flips" the model's prediction and is used for explanation. (Wachter et al., 2017) first propose the concept of unconditional counterfactual explanations and a framework to generate counterfactual explanations. (Hendricks et al.,

2018) first consider the evidence that is discriminative for one class but not present in another class, and learn a model to generate counterfactual explanations for why a model predicts class A instead of B. In this paper, we focus on counterfactual example generation providing contrastive example for natural language inference.

5.2 Post-hoc Explanation Generation

For post-hoc explainable NLP system, we can divide explanations into three types: feature-based, example-based and concept-based.

For feature-based explanation, (Ribeiro et al., 2016) propose LIME and (Guidotti et al., 2018) extend LIME by fitting a decision tree classifier to approximate the non-linear model. However, there is no guarantee that they are faithful to the original model. For example-based explanation, (Kim et al., 2016) select both prototypes and criticisms from the original data points. (Wachter et al., 2017) propose counterfactual explanations providing alternative perturbations. For concept-based explanation, (Ghorbani et al., 2019) explains model decisions through concepts that are more understandable to human than individual features or characters. In this paper, we integrate counterfactual example and concepts for contrastive explanation generation.

5.3 Natural Language Inference

For natural language inference, (Bowman et al., 2015) propose SNLI which contains samples of premise and hypothesis pairs with human annotations. In order to provide interpretable and robust explanations for model decisions, (Camburu et al., 2018) extend the SNLI dataset with natural language explanations of the ground truth labels, named e-SNLI. For explanation generation in NLI, (Kumar and Talukdar, 2020) propose NILE, which utilizes label-specific generators to produce labels along with explanation. However, (Zhao and Vydiswaran, 2021) find NILE do not take into account the variability inherent in human explanation, and propose LIREx which incorporates a rationale enabled explanation generator. In this paper, we consider generating contrastive explanations in NLI.

6 Conclusion

In this paper, we focus on knowledge-aware contrastive explanation generation for NLI. We generate counterfactual examples by changing identified rationales of given instances. Afterwards, we extract concepts knowledge from ConceptNet and dictionary to train knowledge-aware explanation generators. We show that contrastive explanations that specify why a model makes prediction A rather than B can provide more faithful information than other "WHY A" explanations. Moreover, contrastive explanations can be used for data augmentation to improve the performance and robustness of existing model. The exploration of contrastive explanation in other NLP tasks (i.e. question answering) and better evaluation metrics for explanation will be performed in the future.

Acknowledgments

We thank the anonymous reviewers for their helpful comments on this paper. This work is supported by National Key R&D Program of China (No. 2018AAA0101900), the NSFC projects (No. 62072399, No. U19B2042, No. 61402403), Chinese Knowledge Center for Engineering Sciences and Technology, MoE Engineering Research Center of Digital Library, Alibaba Research Intern Program of Alibaba Group, Alibaba-Zhejiang University Joint Institute of Frontier Technologies, and the Fundamental Research Funds for the Central Universities.

References

Pepa Atanasova, Jakob Grue Simonsen, Christina Lioma, and Isabelle Augenstein. 2020. Generating fact checking explanations. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7352–7364, Online. Association for Computational Linguistics.

Samuel R. Bowman, Gabor Angeli, Christopher Potts, and Christopher D. Manning. 2015. A large annotated corpus for learning natural language inference. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 632–642, Lisbon, Portugal. Association for Computational Linguistics.

Oana-Maria Camburu, Tim Rocktäschel, Thomas Lukasiewicz, and Phil Blunsom. 2018. e-snli: Natural language inference with natural language explanations. In *Advances in Neural Information Processing Systems*, volume 31. Curran Associates, Inc.

Qian Chen, Xiaodan Zhu, Zhen-Hua Ling, Si Wei, Hui Jiang, and Diana Inkpen. 2017. Enhanced LSTM for natural language inference. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1657–1668, Vancouver, Canada. Association for Computational Linguistics.

Qianglong Chen, Feng Ji, Haiqing Chen, and Yin Zhang. 2020a. Improving commonsense question answering by graph-based iterative retrieval over multiple knowledge sources. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 2583–2594, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Zhongxia Chen, Xiting Wang, Xing Xie, Mehul Parsana, Akshay Soni, Xiang Ao, and Enhong Chen. 2020b. Towards explainable conversational recommendation. In *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence, IJCAI-20*, pages 2994–3000. International Joint Conferences on Artificial Intelligence Organization. Main track.

Furui Cheng, Yao Ming, and Huamin Qu. 2020. Dece: Decision explorer with counterfactual explanations for machine learning models. *IEEE Transactions on Visualization and Computer Graphics*.

Eunsol Choi, He He, Mohit Iyyer, Mark Yatskar, Wen-tau Yih, Yejin Choi, Percy Liang, and Luke Zettlemoyer. 2018. QuAC: Question answering in context. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2174–2184, Brussels, Belgium. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Amirata Ghorbani, James Wexler, James Y Zou, and Been Kim. 2019. Towards automatic concept-based explanations. In *Advances in Neural Information Processing Systems*, pages 9273–9282.

Riccardo Guidotti, Anna Monreale, Salvatore Ruggieri, Dino Pedreschi, Franco Turini, and Fosca Giannotti. 2018. Local rule-based explanations of black box decision systems. *arXiv preprint arXiv:1805.10820*.

Lisa Anne Hendricks, Ronghang Hu, Trevor Darrell, and Zeynep Akata. 2018. Generating counterfactual explanations with natural language. In *ICML Workshop on Human Interpretability in Machine Learning*, pages 95–98.

Haozhe Ji, Pei Ke, Shaohan Huang, Furu Wei, Xiaoyan Zhu, and Minlie Huang. 2020. Language generation with multi-hop reasoning on commonsense knowledge graph. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 725–736, Online. Association for Computational Linguistics.

Amir-Hossein Karimi, Gilles Barthe, Borja Balle, and Isabel Valera. 2020. Model-agnostic counterfactual explanations for consequential decisions. In *International Conference on Artificial Intelligence and Statistics*, pages 895–905.

Been Kim, Rajiv Khanna, and Oluwasanmi O Koyejo. 2016. Examples are not enough, learn to criticize! criticism for interpretability. In *Advances in Neural Information Processing Systems*, volume 29. Curran Associates, Inc.

Sawan Kumar and Partha Talukdar. 2020. NILE : Natural language inference with faithful natural language explanations. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8730–8742, Online. Association for Computational Linguistics.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online. Association for Computational Linguistics.

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

Ilya Loshchilov and Frank Hutter. 2018. Fixing weight decay regularization in adam. *ICLR*.

Andrew L. Maas, Raymond E. Daly, Peter T. Pham, Dan Huang, Andrew Y. Ng, and Christopher Potts. 2011. Learning word vectors for sentiment analysis. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 142–150, Portland, Oregon, USA. Association for Computational Linguistics.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*, pages 311–318, Philadelphia, Pennsylvania, USA. Association for Computational Linguistics.

Jonathan Pilault, Amine El hattami, and Christopher Pal. 2021. Conditionally adaptive multi-task learning: Improving transfer learning in {nlp} using fewer parameters & less data. In *International Conference on Learning Representations*.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67.

Nazneen Fatema Rajani, Bryan McCann, Caiming Xiong, and Richard Socher. 2019. Explain yourself! leveraging language models for commonsense reasoning. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4932–4942, Florence, Italy. Association for Computational Linguistics.

Karthikeyan Natesan Ramamurthy, Bhanukiran Vinzamuri, Yunfeng Zhang, and Amit Dhurandhar. 2020. Model agnostic multilevel explanations. *arXiv preprint arXiv:2003.06005*.

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

Alon Talmor, Jonathan Herzig, Nicholas Lourie, and Jonathan Berant. 2019. CommonsenseQA: A question answering challenge targeting commonsense knowledge. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4149–4158, Minneapolis, Minnesota. Association for Computational Linguistics.

James Thorne, Andreas Vlachos, Christos Christodoulopoulos, and Arpit Mittal. 2019. Generating token-level explanations for natural language inference. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 963–969, Minneapolis, Minnesota. Association for Computational Linguistics.

Sahil Verma, John Dickerson, and Keegan Hines. 2020. Counterfactual explanations for machine learning: A review. *arXiv preprint arXiv:2010.10596*.

Sandra Wachter, Brent Mittelstadt, and Chris Russell. 2017. Counterfactual explanations without opening the black box: Automated decisions and the gdpr. *Harv. JL & Tech.*, 31:841.

D Randall Wilson and Tony R Martinez. 1997. Improved heterogeneous distance functions. *Journal of artificial intelligence research*, 6:1–34.

Shuyuan Xu, Yunqi Li, Shuchang Liu, Zuohui Fu, and Yongfeng Zhang. 2020. Learning post-hoc causal explanations for recommendation. *arXiv preprint arXiv:2006.16977*.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R Salakhutdinov, and Quoc V Le. 2019. Xlnet: Generalized autoregressive pretraining for language understanding. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc.

Zhuosheng Zhang, Yuwei Wu, Hai Zhao, Zuchao Li, Shuailiang Zhang, Xi Zhou, and Xiang Zhou. 2020. Semantics-aware bert for language understanding. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 05, pages 9628–9635.

Xinyan Zhao and VG Vydiswaran. 2021. Lirex: Augmenting language inference with relevant explanation. *Proceedings of the AAAI Conference on Artificial Intelligence*.

A Appendices

Reported Experimental Results Here, we report some other experimental results for reproduction. We use 2 RTX-6000 GPUs for generator training. For each epoch, it takes 3 hours to fine-tune the contrastive generator. As we set 4 epochs for each "WHY A" generator and "WHY NOT B" generator, it takes 12 hours for each approach. There are 355M parameters in RoBERTa-large, 340M parameters in BERT-large and 345M parameters in GPT2-medium. And our code is based on Pytorch.

The Difference between Counterfactual Example and Contrastive Explanation In this paper, we generate contrastive explanations with qualified counterfactual examples. As counterfactual examples provide example-based explanations, the contrastive explanations provide concept-based explanations and explain "WHY A NOT B". Meanwhile, for end-user, contrastive explanations are easier to understand than counterfactual example, which can integrate external knowledge from knowledge bases.

Common Replaced Words Here, we show some common replaced words in reversal mechanism.

For entailment to neutral, the top 10 removed words are "*man, wearing, white, blue,black, shirt, one, young, people, woman*", the top 10 inserted words are "*people, there, playing, man, person, wearing, outside, two, old, near*". For entailment to contradiction, the top 10 removed words are "*man, wearing, white, blue,black, two, shirt, one, young,people*", the top 10 inserted words are "*people, man, woman, playing,no, inside, person, two, wearing, women*".

For contradiction to neutral, the top 10 removed words are "*wearing, blue, black, man,white, two, red, sitting, young, standing*", the top 10 inserted words are "people, playing, man, woman, two, wearing, near, tall, men, old". For contradiction to entailment, the top 10 removed words are "wearing, blue, black, man,white, two, red, shirt, young, one", the top 10 inserted words are "people, there, man, two, wearing,playing, people, men, woman, outside".

For neutral to entailment, the top 10 removed words are "*white, wearing, shirt, black,blue, man, two, standing,young, red*", the top 10 inserted words are "playing, wearing, man, two, there, woman, people, men, near, person". For neutral to contradiction, the top 10 removed words are "*white, man, wearing, shirt,black, blue, two, standing,woman, red*", the top 10 inserted words are "*woman, man, there, playing,two, wearing, one, men, girl,no*".

The Demand For Contrastive Explanation A "contrastive explanation" explains not only why some event A occurred, but why A occurred as opposed to some alternative event B. Some philosophers argue that agents could only be morally responsible for their choices if those choices have contrastive explanations, since they would otherwise be "luck infested". Moreover, if the answer predicted by a well-trained model is A but confusing with B, it is natural for end-users to ask "why the answer is A rather than B". A similar scenario is possible to occur when a child is going to recognize characters or learn other language skills. Therefore, contrastive explanation generation is essential in critical domains.