Error Causal inference for Multi-Fusion models

Chengxi Li
University of Kentucky
Lexington, KY 40506
chengxili@uky.edu

Brent Harrison
University of Kentucky
Lexington, KY 40506
harrison@cs.uky.edu

Abstract

In this paper, we propose an error causal inference method that could be used for finding dominant features for a faulty instance under a well-trained multi-modality input model, which could apply to any testing instance. We evaluate our method using a well-trained multi-modalities stylish caption generation model and find those causal inferences that could provide us the insights for next step optimization.

1 Introduction

As machine learning models become more complex and training data become bigger, it is harder for humans to find errors manually once some output went wrong. This problem is exacerbated by the black box nature of most machine learning models. When a model fails, it can be difficult to determine where the error comes from. This is especially true in problems that are inherently multimodal, such as image captioning, where often multiple models are combined together in order to produce an output. This lack of transparency or ability to perform a vulnerability analysis can be a major hindrance to machine learning practitioners when faced with a model that doesn't perform as expected.

Recently, more and more people begin to fuse text and visual information for downstreaming task. In many cases, these models utilize specialized, pre-trained models to extract features. In these situations, it is highly likely that the source of these errors is from these pre-trained networks either being misused or not being interpreted correctly by the larger machine learning architecture. In this paper, we explore how one would perform a vulnerability analysis in these situations. Specifically, we are interested in identifying model errors likely caused by these pre-trained networks. Specifically, we aim to diagnose these errors by systematically removing elements of the larger machine learning model to pinpoint what the causes of errors happen to be. This is especially critical in tasks that utilize multi-modality input models since often these models utilize attention. If the model attends to the wrong features, then this error could potentially cascade throughout the network. In other words, we seek to answer the question, "Given a trained model M which has input features x, y, z, if the current test example is not performing well, is that because of the given features or not? If it is, which specific feature is more likely to blame?"

By answering this question, we can give machine learning practitioners, specifically those who are inexperienced with machine learning and AI concepts, some direction in how to improve the performance of their architecture. We summarize our contributions as follows: 1. we provide a practical method to discover causal errors for multi-modality input ML models; 2. we explore how this method can be applied to state-of-the-art machine learning models for performing stylish image captioning; 3. Evaluate our method by through a case study in which we assess whether we can improve the performance of the investigated instance by removing or replacing these problematic features.

2 Related Work

Our approach to sourcing these errors uses causal inference (Peters et al., 2016; Hernán and Robins, 2020). In this section, we will review works related to causal inference as well as works that provided the inspiration for this paper.

Invariance Principle Invariance principle has been used for finding general causal for some outcome under designed treatment process, where people desired to find actual effect of a specific phenomenon. Invariant causal prediction (Peters et al., 2016) has been proposed to offer an practical way to find casuals under linear model assumption. It later got extended to nonlinear model and data (Heinze-Deml et al., 2018). This invariance can be roughly phrased as the outcome Y of some model

M would not change due to environment change once given the cause for this Y. An example of an *environment change* when $Y = M(X, Z, W)$ and the cause for Y is X, could be a change on Z or W. The invariance principle has been popularly used in machine learning models to train causal models (Arjovsky et al., 2019; Rojas-Carulla et al., 2018). We are going to employ the same insight, using the invariance principle to find cause in our paper but landing in different perspectives. We are not intended to train a model, instead, we are going to use the well-trained models to derive the source cause for lower performance instances.

Potential Outcome and Counterfactual. (Rubin, 2005) proposed using potential outcomes for estimating causal effects. Potential outcomes present the values of the outcome variable Y for each case at a particular point in time after certain actions. But usually, we can only observe one of the potential outcome since situations are based on executing mutually exclusive actions (e.g. give the treatment or don't give the treatment). The unobserved outcome is called the "counterfactual" outcome. In this paper we can observe the counterfactual by removing certain input features from the language generation based on multi-input task.

Debugging Errors Caused by Feature Deficiencies This paper is also related to debugging errors from input. While we are more focus on using a causal inference way to get the real cause for low performance rather than only exploring associations (Amershi et al., 2015; Kang et al., 2018)

3 Methodology

The goal of this paper is to perform a causal analysis in order to determine the likely source of errors in a well-trained model. In the following sections, we will outline our underlying hypotheses related to this task and go into details on the task itself.

3.1 Hypothesis

Hypothesis 1: *With a fixed model, if the output of an instance k is unchanged after an intervention, I, then this is called* ***output invariance****. The causes of the output for this instance k are irrelevant to the features associated the intervention, I.*

Using this output invariance principle, we can identify features that are irrelevant to the prediction made. After removing these irrelevant features, the ones that remain should contribute to any errors present in the output. Given the strictness of the output invariance principle, it is often the case that very few features are identified as the cause of any error present. In some cases, no features are identified. In this paper, we are interested in determining the cause of errors by masking out certain features, specifically those that are unlikely to be the cause of an error. As such, we are interested in the specific case where the removal of certain features does not cause the performance of the model to improve. This phenomenon, which we refer to as ***output non-increasing*** will be rephrased below.

Hypothesis 2: *With a fixed model, if the output of an instance, k, after an intervention, I, is either less than or equal to the original performance of instance k, then this is called* ***output non-increasing****. Then, the features associated with intervention, I, are likely irrelevant to the cause of any error.*

In this paper, we specifically perform interventions that involving masking/hammering out certain input features. *Hammering out* features could mean zero out input features or specific weights, or even remove certain input modalities, etc.. In this paper we will change the values of certain input features f to 0. Then, output is regenerated according to this new input. If the output is unchanged (or gets worse), then we will remove this feature f from the causal features list. Before we perform these interventions, we first want to identify the errors which do not relate to any of these features. This leads to the next hypothesis.

Hypothesis 3: *If we hammered out all input features and output invariance still holds for instance k, we will record the cause for instance k having lower performance as being due to model and dataset bias. We will refer to this as* ***bias error****.*

In this paper, we are interested in more than bias errors. With this goal, we arrive at our final hypothesis on performing causal inference for identifying errors.

Hypothesis 4: *If the performance of instance k is poor and the output of instance k is not caused by bias errors, and if all interventions keep feature $f^\star$ unchanged and we still have output non-increasing, we will say $f^\star$ is the error feature which causes the lower performance output for k.*

With all of the above hypotheses we can infer whether the low performance of the instance k is caused by a single feature f or not. Next we will show how these hypotheses can be utilized to identify the causes of errors.

3.2 Causal Graph: with and without Hammering out Features

As we know, when we build a model in deep learning, we always assume a casual graph in advance and then fit data into the graph for training. Figure 1 shows a sample causal graph (a) with multiple input features. These features will be fit into a black box model and finally the model will, in this case, generate some set of output text. Once we have finished training, we will be able to deploy the model and see each testing instance's performance. With a well-trained model, we can perform many interventions, or investigate specific features by intervening on them in different environments. In practice, however, it is impossible for us to obtain all the random environments. Based hypotheses 3 and 4, along with the steps that people take to perform causal predictions in linear (Peters et al., 2016) and non-linear (Heinze-Deml et al., 2018) situations, we, in this paper, give a more detailed and practical definition below to help us identify whether a feature set S is the causal feature set or not when an instance k having error and this error is not a bias error defined in hypothesis 3. Here S could be a feature set composed of a single feature or multiple features. After hamming out some features, we call a remaining feature set P as S's **parental set** when $S \subset P$. We denote $\mathcal{F}_S$ as: $\mathcal{F}_S = \{g(P) \mid P \text{ is a parental set of } S\}$ and

$$g(P) = \begin{cases} P & \text{if } P \text{ satisfies output} \\ & \text{non-increasing,} \\ \emptyset & \text{otherwise.} \end{cases}$$

Then we could extract the estimated causal feature set $\hat{F}$ as:

$$\hat{F} = \bigcap_{F : F \in \mathcal{F}_S} F \tag{1}$$

To simply understand above, we basically check all the parental sets of S on output non-increasing property to finally make decision on whether S is an error casual feature set or not. In this paper, we mainly focus on evaluating single feature set. To better explain, we also display all the interventions (b-h in Figure 1) we have done to the features (masking out some features) when there are a total of 3 features in the assumed causal graph. We will infer the causal feature for a low performance instance k based on all of these potential outcomes before and after interventions. We will use $o_x, x \in \{a, b, c...h\}$ to note the score for output generation of graphs in Figure 1. First, we extract the instances when the error cause is independent of any features where we find all the instances $\mathcal{B}$, which satisfy $o_a == o_h$. Then the following causal feature inferences will exclude detected instances in $\mathcal{B}$ first. As we are specifically interested in single feature errors, we will enumerate the situation when causal features are R, C, S for instance k, respectively. First of all, according to hypothesis 3 and 4, k $\notin$ $\mathcal{B}$. The causal feature is S when: $o_a >= o_b >= o_c >= o_e$; The causal feature is C when: $o_a >= o_c >= o_d >= o_f$; The causal feature is R when: $o_a >= o_b >= o_d >= o_g$.

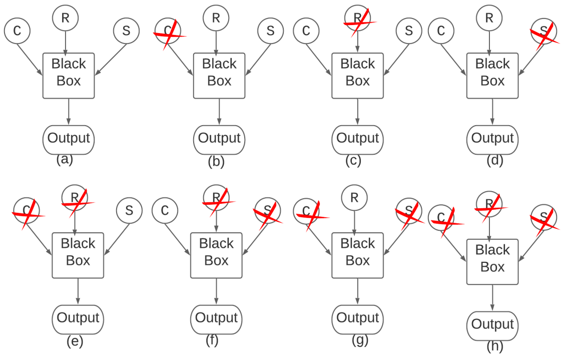

Figure 1: Displays the causal graph with various sets of features zeroed out and a red cross mark signifies zeroing out

It is important to note that removing an error feature does not necessarily mean that the performance will increase, as it is possible that there are other sources of errors that still keep performance low. In these rules, we use the "=" sign in its strictest sense. However, one can always use it in a way that is interchangeable with "very similar." For example, if the difference between two output scores is 10^{-16}, you can choose to regard these two scores as equal per your needs.

4 Experiment

To show the effectiveness of our approach, we will examine its performance on a stylish image captioning task that uses multi-modality feature fusion. While we focus on this task as an example, this approach could be applied to many other domains.

4.1 Dataset

We have chosen the dataset and the task based on three qualities:the work has a well-trained saved model which we could use for intervention and inference; this work still has room to be improved by identifying and removing the source of potential errors; the work utilizes multiple input features in a way that enables removing said features.

Specifically, we choose the work on the 3M model (Li and Harrison, 2021) for stylish text captioning. We do this because it relies on generating captions using several input features including pre-trained text features (C), ResNext features (R), and style information (S) as an input. We would like to explore whether these input features have caused problems when instances are under performing. The dataset we examine is the test set from the PERSONALITY-CAPTIONS dataset (Shuster et al., 2019) using in Li and Harrison's work along with the pretrained model they provide [1]. Even though we use its test set in our experiment, our method could be applied on any set of data of any size when there is a debugging need for multi-fusion models. We will leave this for future work.

4.2 Implementing details

Specifically, we define an instance is under performance in 3M (Li and Harrison, 2021) when the BLEU1 (Papineni et al., 2002) score is lower than the median BLEU1 value among all testing data. In total, we have investigated 9981 instances and 4982 of them are classified as under performing. 74 of these have been detected as bias errors. So finally, 4908 instances have been examined for single feature errors.

We first perform causal inference for style feature and denote those instances that have style error as K_s. Then perform the causal finding steps for ResNext and dense captions without differentiating the order in the remaining instances. The reason to decide such order is due to the structure of 3M, where style is used globally to refine ResNext and dense captions for later stylish text generations while ResNext and dense caption have the same importance for text generations.

4.3 Evaluation

The reason to find the cause for the errors is that we would like to further improve a model when it is well-trained or make a remedy when the model is malfunctioned, especially from the source side. Thus, we evaluate casual predictions by evaluating whether we could improve the model's performance by just altering the causal feature. There are many potential treatments that we could make on the source side such as data augmentation, feature replacement, or feature removal. For each instance k with predicted causal feature f, if its performance could be improved by improving f, then we will judge the causal error inference for this instance k as correct, otherwise incorrect. More details on the specific interventions we use are outlined below: Style: (S1) replace current style with 5 other well-trained styles S, where most instances with style s, $s \in S$ has better BLEU1 score than the medium BLEU1 score. (S2) remove Style. Dense Caption: (C1) replace dense caption to ground truth; (C2) remove dense caption. Resnext: (R1) replace dense caption to ground truth and then remove Resnext, where we make sure at least one of the visual features is valid. (R2) remove Resnext.

We will record the best output BLEU1 score after each intervention. If the intervention results in a higher BLEU1 score than the output prior to the intervention, then the feature in question will be marked as the cause of an error. For all the instances which have been ascribed by a feature f, we calculate the percentage of those in which the BLEU1 score could be improved and report them in Table 1.

5 Result and Discussion

The result is shown in the Table 1. We see that for each feature, most of the instances have increased their performance by improving the predicted features. This performance is also a conservative value as we only did limited feature improvements. For example, for Resnext, we have no better features available and, thus, could not do a replacement. Also in Table 1, the style feature is the most predicted error causal feature among all three feature modalities. We have 1041 instances point its performance error towards style. We speculate this is resulting from the weak training of a certain set of styles, since the BLEU score can be improved if replaced with other better-trained styles for 89.4% of these instances. To further investigate this, we report the frequency of the styles in those 1041 instances in Figure 2 and intend to see whether the estimated error styles are distributed sparsely (all styles are not trained well) or densely (a certain set of styles is not trained well). From Figure 2 we can see that many styles repeatedly appear as errors for various instances, which aligns our speculation. With these predicted error styles, we can either do

[1] https://github.com/cici-ai-club/3M

Table 1: The evaluation result for each feature under casual inference. Predictions Count are the number of instances predicted with corresponding feature errors.

Feature	Predictions Count	Improvement(%)
Style	1041	0.894
Dense Caption	378	0.797
Resnext	300	0.769

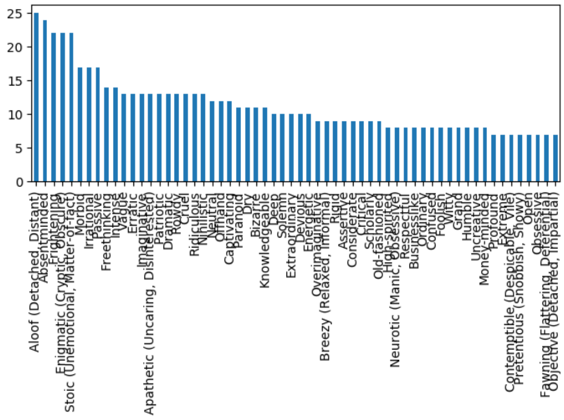

Figure 2: The styles are those frequently predicted as the causal errors; the horizontal bar represents the frequency. Here we select the top 50 styles.

some data augmentation to cover the gap between training and testing or redesign the training process to enable the model to focus more intently on these styles.

6 Conclusion

In this paper, we apply an extended invariance principle to provide a method for performing error causal inference. We evaluate our method under on a stylish image captioning model that uses multi-modal fusion in its input features. We show that we could improve the performance of this model based on simply removing or replacing those found causal errors. Over 70% of the predicted errors could be modified to improve performance. Also, our method is model-agnostic, it could be used for different fusion model for optimization, debugging or assessing purpose.

References

Saleema Amershi, Max Chickering, Steven M Drucker, Bongshin Lee, Patrice Simard, and Jina Suh. 2015. Modeltracker: Redesigning performance analysis tools for machine learning. In *Proceedings of the 33rd Annual ACM Conference on Human Factors in Computing Systems*, pages 337–346.

Martin Arjovsky, Léon Bottou, Ishaan Gulrajani, and David Lopez-Paz. 2019. Invariant risk minimization. *arXiv preprint arXiv:1907.02893*.

Christina Heinze-Deml, Jonas Peters, and Nicolai Meinshausen. 2018. Invariant causal prediction for nonlinear models. *Journal of Causal Inference*, 6(2).

Miguel A Hernán and James M Robins. 2020. Causal inference: what if.

Daniel Kang, Deepti Raghavan, Peter Bailis, and Matei Zaharia. 2018. Model assertions for debugging machine learning. In *NeurIPS MLSys Workshop*.

Chengxi Li and Brent Harrison. 2021. 3m: Multi-style image caption generation using multi-modality features under multi-updown model.

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 on association for computational linguistics*, pages 311–318. Association for Computational Linguistics.

Jonas Peters, Peter Bühlmann, and Nicolai Meinshausen. 2016. Causal inference by using invariant prediction: identification and confidence intervals. *Journal of the Royal Statistical Society. Series B (Statistical Methodology)*, pages 947–1012.

Mateo Rojas-Carulla, Bernhard Schölkopf, Richard Turner, and Jonas Peters. 2018. Invariant models for causal transfer learning. *The Journal of Machine Learning Research*, 19(1):1309–1342.

Donald B Rubin. 2005. Causal inference using potential outcomes: Design, modeling, decisions. *Journal of the American Statistical Association*, 100(469):322–331.

Kurt Shuster, Samuel Humeau, Hexiang Hu, Antoine Bordes, and Jason Weston. 2019. Engaging image captioning via personality. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 12516–12526.