Counterfactual-Based Synthetic Case Generation

Chapter · June 2024

DOI: 10.1007/978-3-031-63646-2_25

Anik Sen
Premier University
20 PUBLICATIONS **121** CITATIONS
SEE PROFILE

Mallika Mainali
Drexel University
4 PUBLICATIONS **2** CITATIONS
SEE PROFILE

Christopher Rauch
Drexel University
12 PUBLICATIONS **12** CITATIONS
SEE PROFILE

Prateek Goel
Drexel University
8 PUBLICATIONS **26** CITATIONS
SEE PROFILE

Counterfactual-Based Synthetic Case Generation

Anik Sen[1], Mallika Mainali[1], Christopher B. Rauch[1], Ursula Addison[2], Michael W. Floyd[3], Prateek Goel[1], Justin Karneeb[3], Ray Kulhanek[2], Othalia Larue[2], David Ménager[2], Matthew Molineaux[2], JT Turner[3], and Rosina O Weber[1]

[1] Drexel University, Philadelphia, PA 19104, USA,
as5867@drexel.edu
[2] Parallax Advanced Research, 4035 Colonel Glenn Hwy, Beavercreek, OH 45431
[3] Knexus Research, 174 Waterfront Street, Suite 310, National Harbor, MD 20745

Abstract. Case augmentation is often desirable when applying case-based reasoning to real-world problems. Initially explored for explainability, counterfactuals were recently recommended as a strategy to augment data. In this work, we implement an existing approach for generating counterfactuals, propose one variant of the original approach, and propose a third approach based on the literature on algorithmic recourse. We apply these three approaches to two datasets in military medical triage. To assess generalization, we also examine one of our approaches on three publicly available datasets. We compare the approaches based on the number of counterfactuals they produce, their resulting accuracy, overlapping counterfactuals, and domain knowledge. Experimental results are encouraging for the proposed approaches and bring up opportunities for future research.

Keywords: Counterfactuals, Counterfactual Generation, Synthetic Cases, Case-Based Reasoning (CBR), Data Augmentation, Synthetic Data

1 Introduction

Case-based reasoning (CBR) leverages past experiences to solve new problems [11]. Despite its usefulness in various contexts, limited data poses challenges for CBR. One domain where data is often limited is in medical decision-making, due to privacy and ethical concerns [12]. Within computer science, counterfactuals (CFs) have extensive use for explainable artificial intelligence (XAI).

Recently, Temraz and Keane [14] demonstrated the use of Keane and Smyth's [5] approach to CF generation to attenuate problems with class imbalance. CF-based synthetic case generation relies on the factual-CF relationship between cases to identify regions of the data space that may be filled with new instances. In this paper, we limit the scope of our examination to CF-based synthetic case generation for data augmentation.

When applying Keane and Smyth's [5] approach for augmenting medical triage cases, the number of generated synthetic cases was insufficient for our needs. For this reason, we explored variations of said approach. In this paper, we implement Keane and Smyth's [5] approach, which we henceforth refer to as

Keane's-CF, analyze it, and compare it with two other approaches. Approach 2 is inspired by Keane's-CF; we call it Direct-CF. Approach 3, denoted by High-Weights-CF, is a novel approach inspired by research in algorithmic recourse (*e.g.*, [3, 6]). This paper's intended contribution is to present an analysis of these three approaches.

The analysis methodology we employ anticipates that all approaches receive as input an *original case base* with an average accuracy previously recorded via leave-one-out cross-validation (LOOCV). The new cases generated by the approaches are then used to build a new case base that we test by assessing its ability to classify the cases from the *original case base*. The average accuracy of the synthetic cases is one of the aspects used in the analysis. Other aspects include a further ablation of those cases to determine if the quality of individual cases differs, the number of synthetic cases generated, whether the generated cases overlap, and an analysis of their plausibility based on domain knowledge.

In the next section, we describe related work. Section 3 describes the three approaches we analyze. Section 4 presents the experimental design. Next are results and discussion, then conclusion in Section 6.

2 Related Work

CFs were originally proposed to provide explanatory information in XAI to users (*e.g.*, [17, 4, 16, 20, 21]). Given the relationship between a factual and its CF, generating synthetic CFs for data augmentation is a suitable approach. Data augmentation based on CF generation [10, 2, 14, 1] using actual feature values rather than interpolating between instances have been demonstrated to increase the number of plausible cases [14].

Keane and Smyth highlighted the challenges in finding *good* CFs and proposed a case-based technique to generate plausible CFs by reusing the patterns of good CFs present in a case base [5]. This approach generates CFs based on similarity metrics rather than random perturbations so that the CFs are likely to be inherently plausible and sparse. Similarly, to generate sparse and diverse CFs, Smyth and Keane proposed a method to adapt native CFs existing in the original dataset to generate synthetic CFs from naturally occurring features [13].

CFs are the basis of algorithmic recourse (AR) where the distinction lies in that AR requires changes made to the instances to be *feasible* [15]. Karimi et al. highlighted the inadequacy of CF explanations in offering actionable recommendations capable of positively altering model predictions due to a lack of consideration of causal relations [3]. To address this limitation, they reformulated the framework proposed by Ustun et al. [15] by adding a plausibility constraint to generate optimal CFs. Building upon this foundation, Konig et al. introduced a method to constrain AR to recommend only actions deemed *meaningful*, which aimed to improve both the prediction of the model and the target [6]. They proposed that by generating causal instances, the resulting CFs would be meaningful, ensuring that the algorithm's performance remains intact.

Conversely, generating instances that lack causality would not provide a guarantee that the resulting CFs lie within the data manifold, thereby jeopardizing the classifier's predictability [16]. Taking all these factors into consideration, O'Brien et al. devised a novel algorithm called Causal Augmented Sparse Classification Algorithm to implement causality and generate CFs solely by using causal features [8]. Given the implementation of causality requires ground truth and domain rules, which were unavailable in our synthetic cases, we resorted to the next best approach: correlation. In our High-Weights-CF approach, we use correlation indicated by feature weights to prioritize the features to be altered while generating CFs, as elaborated in Section 3.3.

3 Three Approaches for Synthetic Case Generation

Common to the approaches we describe next is the identification of *native* and *non-native* CF sets. A native CF is a pair of cases where one is the *factual* and one is its CF (*i.e.*, similar but with a different decision label). The term

Algorithm 1: Pseudocode for Keane's-CF approach.

```
Input: Case base
Output: A set of synthetic cases
similarity_threshold, feature_difference ← set similarity threshold
  and feature difference
native_cf ← [], non_native_cf ← []
for case1 in Case base do
    Compare case1 with the other cases
    Calculate local similarity, global similarity, and feature difference
      between case1 and all the other cases
    Create a list for case1 with all the retrieved cases
    Put case1 in the non_native_cf if it has no native CF (within
      threshold). Otherwise put case1 in the native_cf
new_cases ← []
for case_non_cf in non_native_cf do
    for case_cf in native_cf do
        global_sim ← Calculate global similarity between case_non_cf and
          case_cf
        if global_sim ≥ similarity_threshold then
            similar_cases ← Find all of the similar cases of case_cf
            for sim_case in similar_cases do
                if sim_case has different label than case_non_cf then
                    diff_attr_names ← Find which attributes have
                      different values between case_cf and sim_case
                    sim_attr_names ← Set the remaining attributes as
                      similar
                    new_case ← Copy diff_attr_names from sim_case
                      and sim_attr_names from case_non_cf
                    Set label of sim_case to new_case
                    if new_case Not Exists In Case base then
                        Append new_case to new_cases
return new_cases
```

native expresses that the pair occurs naturally in the data rather than it is created artificially. *Non-native* CF pairs indicate sets of cases for which there are no native CFs naturally occurring in the data for the factual cases. These definitions are based on the previous usage of the term [14] but in this paper we use a looser definition of similar from that used before [5, 14].

3.1 Keane's-CF Approach

This approach is mostly based on previous work by Keane and Smyth [5], with two differences. First, we relax the notion that a CF has only two feature differences by exploring CFs with more than two feature differences. Second, we do not submit the synthetic CF to the model for classification. Instead, we simply reuse the label. For these differences, this is not really the same as Keane and Smyth approach; though the remaining steps are the same. In Algorithm 1, we outline the process of generating synthetic cases based on a similarity threshold and feature differences. We compile two sets of cases: native CFs (*native_cf*) and non-native CFs (*non_native_cf*) (lines 3-7). We then iterate through each case in *non_native_cf*, comparing it with every case in *native_cf* to assess similarity

Algorithm 2: Pseudocode for Direct-CF approach.

```
   Input: Case base
   Output: A set of synthetic cases
1  similarity_threshold, feature_difference ← set similarity threshold
     and feature difference
2  native_cf ←[], non_native_cf ←[]
3  for case1 in Case base do
4      Compare case1 with the other cases
5      Calculate local similarity, global similarity, and feature difference
         between case1 and all the other cases
6      Create a list for case1 with all the retrieved cases
7      Put case1 in the non_native_cf if it has no native CF for a certain
         threshold. Otherwise put case1 in the native_cf
8  new_cases ← []
9  for case in non_native_cf do
10     non_cf_cases ← Find all cases with similar labels within a certain
         feature difference.
11     Find a native CF case (native_cf) of case with the lowest feature
         difference
12     different_attributes ← Compute which attribute values are
         different between case and native_cf
13     for case2 in non_cf_cases do
14         Create case (new_case) by taking the attribute from native_cf
             that differ between case and native_cf and remaining attributes
             from case2
15         Make the label of native_cf the label of new_case
16         if new_case Not Exists In Case base then
17             Append new_case to new_cases
18 return new_cases
```

between them (lines 13-14). For similar cases found, we first take the CF of the similar case and identify varying attributes between this CF and the case from the non_native_cf set. We then create a CF case based on attributes from both the case from the non_native_cf set and the CF of the similar case ($case\_non\_cf$). The label of the new case corresponds to the similar case's (sim_case) label. To clarify the attributes used, we classify the features of the two cases as *different-features*, denoting features with distinct values in them, and *match-features*, representing features with identical values. Then, we create the synthetic CF by transferring the *different-features* values from the native CF and *match-features* from the case from the non_native_cf set.

3.2 Direct-CF Approach

The intuition behind this approach is that when cases have a lot of similar cases that include both CFs and similar cases with the same label, that the variability of similar cases with the same label could be used to increase the opportunities of creating synthetic CFs than by creating synthetic CFs indirectly by using the CF of a similar case,as in Keane's approach. This is why we call this *Direct-CF*.

Algorithm 2 starts by creating native CF ($native\_cf$) and non-native CF (non_native_cf) sets (Lines 3-7). Then, for each case ($case$ in line 9) in the second set, we find the first occurrence of a CF case ($native\_cf$ in line 11) to extract attributes with the lowest feature difference (line 12). Subsequently, we iterate through each case in the second set ($case2$ in line 13) to identify similar cases. A new case is generated by combining feature values from $native\_cf$ and similar attribute values from $case2$. The label of the new case follows that of $native\_cf$. This new case is considered if it does not already exist in the case base. The process is outlined in Fig. 1.

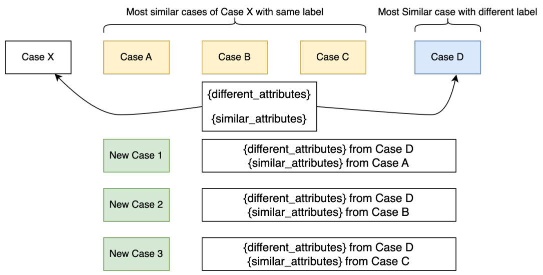
Most similar cases of Case X with same label
Most Similar case with different label
Case X
Case A
Case B
Case C
Case D
{different_attributes}
{similar_attributes}
New Case 1
{different_attributes} from Case D
{similar_attributes} from Case A
New Case 2
{different_attributes} from Case D
{similar_attributes} from Case B
New Case 3
{different_attributes} from Case D
{similar_attributes} from Case C

Fig. 1. Visualization of the process of Direct-CF approach.

3.3 High-Weights CF approach

Our third approach, as outlined in Algorithm 3, is based on the feature importance of the attributes. Based on the notion that causal features are better choices of features to change to obtain a different label, as discussed in Section 2, in lack of which features are causal, we focus on the features that are likely to have higher correlations to the label, those with highest weights to consider differences between cases as basis of CFs. By high weights, we mean weights above a set threshold. The remainder of the process is similar to Keane's-CF.

Algorithm 3: Pseudocode for High-Weights-CF approach.

```
   Data: Case base
   Result: A set of synthetic cases
1  $similarity\_threshold$, $feature\_difference$, $weight\_threshold \leftarrow$ set
     similarity threshold, feature difference, and weight threshold
2  $native\_cf$, $non\_native\_cf \leftarrow$ [], []
3  $feature\_importance \leftarrow$ Extract feature importance of the attributes by
     executing the weight learning algorithm
4  for case1 in Case base do
5      Compare $case1$ with the other cases
6      Calculate local similarity and global similarity between $case1$ and all
         the other cases
7      Computer feature differences for those attributes where the value is
         greater than $feature\_importance$
8      Create a list for $case1$ with all the retrieved cases
9      Put $case1$ in the $non\_native\_cf$ if it has no native CF. Otherwise
         put $case1$ in the $native\_cf$
10 (Lines 8-22 are same as Algorithm 1)
```

4 Experimental Design

Given a dataset, we implement preprocessing, case base creation and evaluation, then execute case augmentation and evaluate synthetic cases, and finally evaluate the resulting case base after adding augmented cases. The next sections describe these steps.

4.1 Datasets

We use five datasets. Two datasets, A and B, describe decisions of medical triage. The other three datasets are from the UCI repository[4], which we included to assess generality of the approaches. The characteristics of the datasets are presented in Table 1.

Datasets A and B entail decisions collected from medical decision-makers presented with battlefield triage scenarios. These scenarios probe the decision-making process in austere combat situations where medics must quickly assess

[4] The UCI repository is available at: https://archive.ics.uci.edu/datasets

Table 1. Characteristics of the datasets used in the study.

Dataset	Instances	Features	Unique Labels	Feature Types	Missing Values
A	792	169	68	Binary, Categorical, Numeric	Yes
B	4138	141	74	Binary, Categorical, Numeric	Yes
Votes	435	16	2	Categorical	Yes
Ecoli	336	7	7	Binary, Categorical, Continuous	No
Soybean	307	35	19	Categorical	Yes

the condition of wounded soldiers and decide on the most appropriate course of action, such as immediate treatment, evacuation priority, or deferment of care based on the severity of injuries and available resources. These datasets were created by performers in the ITM DARPA Project [7].

Cases represent a combination of scenario features, supplemented features [19] added by decision analysis [7], and a decision. We consider these cases as ground truth, as they are developed with domain experts. Next are descriptions of some of the important features of this data.

Action: decision chosen by medical decision-makers in response to triage scenarios. Possible values include *apply treatment, check all vitals, check pulse, check respiration* and *move to evacuation.*

Supplies: medical resources and equipment available to carry out treatments. Examples of possible values are *tourniquet, pressure bandage, hemostatic gauze, burn dressing, and nasopharyngeal airway.*

Treatment: particular medical intervention selected as a result of the *action* attribute. It is based on the administration of an available supply.

Severity of injury: is categorized from low to extreme, indicating the urgency of medical attention needed.

Casualty tag: *tagging* a casualty involves assessing each individual's injuries or condition to give a tag that indicates the priority for receiving medical treatment. Tag values range from *minor*, which requires minimal care, to *expectant*, which indicates survival is unlikely even with optimal medical resources.

Soldier's status: including combat role, vital signs, and potential for recovery with immediate care.

Environmental factors: such as the proximity of ongoing combat, weather conditions, the likelihood of safely evacuating the wounded, and the delay of delivering aid.

Decision Analytics: are data generated from the observed scenario using a variety of algorithms. The values are not themselves observed data but represent analytical results produced by the application of evaluative metrics, such as the probable severity change of an injury over time, or the probabilities of events such as death by asphyxia, depending on the values in the dataset [7].

4.2 Preprocessing

Preprocessing addresses missing values, attributes lacking distinct values, and duplicate samples. Initially, we replace all empty values with -999. Attributes with fewer than ten distinct values are considered categorical, while those with ten or more distinct values are treated as numerical. We use this unless available domain rules dictate a different scale. We calculate the range between the highest and lowest values for numerical attributes as a measure of feature variability. This feature variability is used to assess the local similarity between cases and informs subsequent steps in the CF generation process.

4.3 Create and Evaluate Case Bases

To create case bases from data, we define similarity assessment through local similarity functions, learning feature weights, and global similarity for aggregation. We calculate local similarity between two cases by using functions based on their feature type. For categorical features, the algorithm checks for exact equality. For numerical features, similarity is calculated based on the absolute difference between the feature values, scaled by the difference of maximum and minimum values occurring in the data for that feature. The process of calculating local similarity is highlighted in Algorithm 4. Then, we use a weight learning method to obtain weights for each attribute. To compute the global similarity score between an input case and candidate cases, we aggregate local similarity and weights with the weighted mean.

Algorithm 4: Local similarity calculation between two cases.

```
Input: case1, case2, feature type
Output: Similarity between two cases
local_sim ← [], $i \leftarrow 0$
for $c1, c2$ in zip(case1, case2) do
    if $feature\_type[i][0]$ is Categorical then
        if $c1$ is equal to $c2$ then
            append 1 to local_sim
        else
            append 0 to local_sim
    else
        $temp \leftarrow 1 - \left(\frac{|c1-c2|}{feature\_type[i][1]}\right)$
        append temp to local_sim
    $i \leftarrow i + 1$
return local_sim
```

Once we have a case base, then we assess the average accuracy using LOOCV.

4.4 Case Augmentation and Evaluation of Synthetic Cases

This is where we execute algorithms 1, 2, and 3 from Section 3. They all use a case base as input. Once an approach produces augmented cases, we evaluate the quality of the augmented cases by assessing how well they predict the original case base (*i.e.*, using original cases as a test set).

4.5 Combined Evaluation of Original and Augmented Cases

Average Accuracy. We evaluate each approach and each feature difference. To assess the quality of the synthetic cases, we predict the cases in the original case base by considering all the newly generated synthetic cases as the training samples. We only consider testing cases from the case base if the labels are present in the new cases, as we do not generate new cases for all labels. After these results, we conduct ablation studies, which we detail with the results. For comparison and comprehension of the value of the approach, we evaluate the overall accuracy of the resulting case base after adding all the synthetic cases to the original case base, which was our original goal.

Overlap analysis. Overlap analysis compares the cases generated by each approach to check whether different approaches would produce the same case.

Plausibility analysis. The evaluation method used to assess the plausibility of counterfactual cases generated in our study focused on ensuring that each data point adhered to certain logical rules reflecting the domain. This systematic checking of data plausibility involved both automatic and manual reviews.

Automatic validation processes ensured that the data points conformed to predefined constraints derived from the nature of the data. For instance, in the medical triage context of the examined probes, specific treatments are only applicable to certain conditions or anatomical locations, and a treatment cannot be applied if the required supplies are unavailable. Given the discrete choices of data on actions, treatments, and supplies, the categorical components of the dataset, these checks could be programmatically enforced, ensuring that each data point was plausible within the operational constraints of the dataset. This method reflects a common practice in synthetic data generation, where constraints are defined based on domain knowledge to ensure the generated data maintains logical consistency. In addition to automated checks, samples of the data were manually reviewed to ensure they adhered to common sense and exhibited logical consistency. This manual checking involved assessing if the treatment and actions applied were plausible and did not result in contradictory or impossible actions. Such manual review provides an additional layer of assurance that the automated processes did not overlook non-feasible or implausible results.

Generalizability. The datasets from the UCI repository are evaluated to assess the generality of the approaches, as described in Section 5.4. As the UCI datasets have fewer attributes as compared to our triage datasets, we restrict the analysis to two and three feature difference. We measure performance using the number of cases generated and average accuracy (Acc.) in predicting the cases with similar labels. For the Ecoli dataset, we restrict our analysis to single precision floating point values. Initially, we partition the case base into non-CFs and CFs by employing feature differences of two and three. Subsequently, in Direct-CF approach, to identify the most similar CF case for each non-CF case, we incrementally adjust the feature difference by one unit, thereby ensuring the plausibility of the generated new cases.

5 Results and Discussion

5.1 Accuracy

The results of three different approaches for two datasets are shown in Table 2. We consider the number of new cases generated and average accuracy (Acc. %). We observe that the new cases are created only for certain labels. While calculating the average accuracy, we only predict those instances that have the same labels as the original cases. We compare the performance for two, three, and four feature differences. There are times when neither dataset generates a new case, making it impossible to calculate the average accuracy. Keane's-CF records the highest, 100% accuracy using a feature difference of two for Case Base A. This approach records only 35.29% accuracy using three feature differences on Case Base B. For Case Base A, Direct-CF is 92.74% accurate using both two and three feature differences, while for Case Base B, the highest accuracy is 95.85%. High-Weights-CF achieves 100% accuracy for Case Bases A (three feature differences) and B (four feature differences). Direct-CF generated a good number of cases for all feature differences. For the third approach, there is an additional parameter, feature importance, for which we set the value 0.015. It is also apparent that Direct-CF generates more cases than the other two approaches. That is because it is considering one most similar CF case for each non-CF case. For some non-CF cases, the most similar case has a higher number of feature differences and for some cases, the most similar case has fewer feature differences with high similarity.

The original case base accuracy for both the triage datasets are shown in second column (Base Acc. %) of Table 2. Then, we calculate the overall accuracy of both case bases, including all the new cases. Keane's-CF enhances the accuracy of Case Base A from 87.25% to a maximum of 88% with two feature differences, while there is no improvement for Case Base B. Direct-CF improves the overall accuracy for both case bases across all feature difference values. For Case Base A, accuracy increases to 89.60% with two feature differences, and for Case Base B, it rises to 99.98% with four feature differences. High-Weights-CF increases the accuracy of Case Base A to 89.66% with three feature differences and Case Base B to 98.68% with four feature differences.

Although the Keane's-CF and High-Weights-CF approaches produced a limited number of CF cases, employing these methodologies can still be advantageous. By presenting a broad spectrum of scenarios for the model to learn from, these techniques may significantly improve its ability to generalize across heterogeneous datasets. The absence of case overlap in our experiments underscores the complexity of integrating various augmentation techniques into a comprehensive augmentation strategy. However, it also highlights the value of experimenting with different approaches, as some may be more effective than others depending on the specific characteristics of the data in question.

Table 2. Results of three different approaches on two case bases.

			Keane's-CF		Direct-CF		High-Weights-CF	
Case Base	Base Acc. (%)	Exact. Feat. Diff	New Cases	Acc. (%)	New Cases	Acc. (%)	New Cases	Acc. (%)
A	87.25	2	1	100	131	87.06	8	81
		3	4	33.33	141	92.74	1	100
		4	None	N/A	142	92.74	None	N/A
B	98.65	2	None	N/A	1071	90.71	None	N/A
		3	6	35.29	1381	88.93	None	N/A
		4	None	N/A	1733	95.85	90	100

Table 3. Results for ablation study by randomly dropping new cases using Direct-CF from Dataset A. The results are based on five iterations.

New Cases	Drop Rate.	Max Acc. (%)	Min Acc. (%)
131	10%	95.4	86.73
	20%	95.1	84.62
141	10%	94.7	89.8
	20%	92.7	88.8

Keane's-CF for Case Base A creates four cases. In an ablation study, we remove each instance and compare average accuracy. Removing the first case improves accuracy to 66.67%. Removing the second, third, and fourth cases individually does not change average accuracy. In the second ablation study, as highlighted in Table 3, we randomly drop 10% and 20% of the cases generated by Direct-CF from Case Base A (131, 141). Although there is evidence of a 20% reduction in cases, the average accuracy increases to 94% for some experiments. These two ablation studies suggest further refinement of the newly generated cases.

5.2 Overlap Analysis

This section describes the overlapping CFs resulting from three approaches. We observed that each approach produced a distinct set of new cases. A comparison between the new cases generated from Case Base B by Direct-CF and High-Weights-CF, considering a feature difference of four, revealed that High-Weights-CF produced 29 new cases with label 35 and 61 new cases with label 36. On the other hand, among the 1733 new cases produced by Direct-CF, 101 were labeled as 35 and 130 as 36. Remarkably, no overlaps or subsets were found between the datasets from the two approaches. This was also the case with Case Base A, where we examined the new cases generated by each approach for feature differences two and three and discovered that none of the approaches had overlapping cases for any number of feature differences. The lack of overlap suggests that each approach generates distinct cases.

5.3 Domain Analysis and Plausibility

Plausibility in CFs is defined as the realism and relevance of these hypothetical alternatives to actual events or data points. For CFs to be considered plausible, they must meet several criteria, including relevance to the domain, adherence to causal relationships presented in the empirical data, and the ability to be actionable or achievable under the constraints set by the nature of the case. The synthetic examples should also reflect the underlying distribution and characteristics of the original datasets [9, 18].

In our experiments, we generate CFs from the reference datasets developed by domain experts. When these datasets are used as a basis for generating CFs, our initial assumption is that they accurately represent observed real-world scenarios, patterns, and distributions.

The original datasets capture decisions made under a variety of conditions and have native CF cases. By methodically varying between two and four attributes of these cases, such as the severity of injuries, availability of resources, or environmental factors, new CF cases are constructed for those cases in the original case base that do not already have them. In order to evaluate the plausibility of these CFs without specific domain expertise, we examine four criteria:

Consistency with the Reference Data: CFs should maintain consistency with the patterns, relationships, and distributions found in the reference datasets. This ensures that the CF scenarios or augmented data points they represent are within the realm of possibility for the given domain.

Adherence to Known Causal Relationships: Even without domain-specific knowledge, CFs can be assessed for their plausibility based on their adherence to causal relationships that are either evident within the dataset or established through general knowledge. This involves evaluating whether the changes proposed by CFs make logical sense given the observed cause-and-effect dynamics in the original data.

Feasibility and Actionability: CFs should propose changes or scenarios that are actionable within the constraints of the domain, even as understood by a non-expert. For example, if a limb has been amputated, a treatment that requires the application of a supply to that (nonexistent) limb is not actionable or feasible.

Alignment with Domain Goals and Values: Even without in-depth domain expertise, the evaluation of CFs can consider their alignment with the overarching goals, values, and ethical standards of the domain. This includes the obvious criteria that the application of treatment should not cause harm but does not extend to the trade off between harm and mission accomplishment. Here, we mean obvious harm, such as denying a patient care completely rather than prioritizing the order of care.

We analyze the plausibility of example CFs created from the case base, as shown in Table 4. It would be tempting to present a comparison of original cases and their CFs showing the feature change of two or three and thereby inferring a causal relationship of those features directly to the change in the decision. However, the high dimensionality of the source data complicates the direct identification of specific feature differences responsible for the transition from one decision outcome to the next. The heterogeneity of intra-feature correlations that are involved in the triage medical dataset further obscures the relationship between specific features and changes in the predicted action and treatment. Generated cases are created by altering the original data are not real-world observations themselves, so we can only discuss correlation, not causation, as we have done in analyzing feature weights for class prediction. Additional observations are as follows. The CF is consistent with the reference dataset. The CF generated appears as an available treatment in the original dataset. The adherence to the causal relationship known from the base data is also preserved by virtue of its existence in the original dataset. There is no obvious misalignment with the domain goals and values. For example, no CF case contains an instance of administering pain medication to someone who is not injured. The result is actionable. For example, the suggested case does not create a situation that would not be feasible given the domain knowledge represented by the original case base. Possible features that are not actionable would be those that defy logic; for example, applying a treatment such as a tourniquet to *left face*.

Table 4. An example case and a CF represented in abbreviated form. The relationship between the original and CF cases will not necessarily be directly apparent from the feature differences, but the derived case should be plausible.

Base Case	name: TAG_CHARACTER, params: (category: MINIMAL, casualty: CASUALTY_X), environment: DESERT, outcome_probability: 0.93
CF	name: APPLY_TREATMENT, params: (treatment: NASOPHARYNGEAL AIRWAY, casualty: CASUALTY_V, location: LEFT FACE), environment: JUNGLE, outcome_probability: 0.95

5.4 Performance on Other Datasets

The results are presented in Table 5. It is apparent that Keane's-CF performs better for all three datasets in predicting the cases while using the new cases as the training samples. The base accuracies of the Votes, Ecoli, and Soybean datasets are 95.40%, 73.51%, and 91.53%, respectively. We analyze whether the new cases increase base accuracy. None of the approaches record an accuracy increase for the Votes dataset. Keane's-CF, using three feature differences, increases the base accuracy of the Ecoli dataset to 90.39%. It also increases the base accuracy of the Soybean dataset to 92.81% using two feature differences. Direct-CF and High-Weights-CF both record about a 1% accuracy increase using two and three feature differences.

While the Direct-CF approach produces more cases for the triage datasets, it exhibits different behavior for the UCI datasets, as illustrated in Table 5. UCI

Table 5. Performance of the three approaches on three UCI datasets.

Dataset	**Base. Acc. (%)**	**Exact Feat. Diff.**	**Keane's-CF**		**Direct-CF**		**High-Weights-CF**	
			New Cases	**Acc. (%)**	**New Cases**	**Acc. (%)**	**New Cases**	**Acc. (%)**
Votes	95.40	2	1340	80.92	None	N/A	None	N/A
		3	6389	72.18	None	N/A	None	N/A
Ecoli	73.51	2	363	77.51	104	38.90	402	70.82
		3	2141	87.72	93	51.09	1975	59.28
Soybean	91.53	2	13	49.17	60	41.67	3	2.00
		3	238	65.38	66	54.44	40	64.29

datasets are standardized to some extent, whereas our medical triage datasets require additional standardization techniques. Furthermore, we acknowledge that each approach may behave differently based on the nature of data and preprocessing techniques. Further exploration will provide insight into these behaviors.

6 Conclusion and Future Work

In this research, we proposed two novel approaches based on an established framework [5] for generating CF-based synthetic cases. The approaches help to address data scarcity issues often present in the medical domain. For the two medical triage datasets, our Direct-CF approach demonstrates better results in predicting the cases in the original case base by taking all the new cases as the training set. This approach also increases overall accuracy by combining new cases with original cases. Despite the low number of CFs generated by two of the approaches using triage datasets, the exploration of alternative approaches produces synthetic data with high levels of accuracy.

Our studies reveal opportunities for future research. It is clear that not all new cases generated are equally accurate, but without close examination and consideration of domain knowledge, we cannot automatically distinguish which ones to keep. High-Weights-CF approach was conceived based on the notion that correlation is the closest we can come in the absence of ground truth for causality. In future work, it would be beneficial to examine how feature weights may affect the consideration of feature differences.

Acknowledgements This research was conducted as part of the In the Moment (ITM) project, supported by the Defense Advanced Research Projects Agency (DARPA) under contract number HR001122S0031. We acknowledge other performers from ITM who provided the datasets in the military medical triage. Dataset A was based on data provided by Alyssa Tanaka and colleagues with Soar Technology, LLC. Dataset B was based on work provided by RTX BBN Technologies.

References

1. Gowtham Reddy, A., Bachu, S., Dash, S., Sharma, C., Sharma, A., Balasubramanian, V.N.: Rethinking counterfactual data augmentation under confounding. arXiv e-prints pp. arXiv–2305 (2023)
2. Hasan, M.G.M.M., Talbert, D.A.: Counterfactual examples for data augmentation: A case study. In: The International FLAIRS Conference Proceedings, vol. 34 (2021)
3. Karimi, A.H., Schölkopf, B., Valera, I.: Algorithmic recourse: from counterfactual explanations to interventions. In: Proceedings of the 2021 ACM conference on fairness, accountability, and transparency, FAccT '21, p. 353–362 (2021)
4. Keane, M.T., Kenny, E.M., Delaney, E., Smyth, B.: If only we had better counterfactual explanations: Five key deficits to rectify in the evaluation of counterfactual xai techniques. In: Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI-21, pp. 4466–4474 (2021)
5. Keane, M.T., Smyth, B.: Good counterfactuals and where to find them: A case-based technique for generating counterfactuals for explainable ai (xai). In: Case-Based Reasoning Research and Development: 28th International Conference, ICCBR 2020, Salamanca, Spain, June 8–12, 2020, LNAI 12311, pp. 163–178. Springer (2020)
6. König, G., Freiesleben, T., Grosse-Wentrup, M.: A causal perspective on meaningful and robust algorithmic recourse. arXiv preprint arXiv:2107.07853 (2021)
7. Molineaux, M., Weber, R., Floyd, M.W., Menager, D., Larue, O., Addison, U., Kulhanek, R., Reifsnyder, N., Rauch, C., Mainali, M., Sen, A., Goel, P., Karneeb, J., Turner, J., Meyer, J.: Aligning to human decision-makers in military medical triage. In: D.B. Juan A. Recio Garcia Mauricio G. Orozco-del-Castillo (ed.) ICCBR 2024, Lecture Notes in Computer Science. Springer (2024)
8. O'Brien, A., Kim, E., Weber, R.: Investigating causally augmented sparse learning as a tool for meaningful classification. In: 2023 IEEE Sixth International Conference on Artificial Intelligence and Knowledge Engineering (AIKE), pp. 33–37. IEEE (2023)
9. Par, Ö.E., Sezer, E.A., Sever, H.: Small and unbalanced data set problem in classification. In: 2019 27th Signal Processing and Communications Applications Conference (SIU), pp. 1–4 (2019)
10. Pitis, S., Creager, E., Garg, A.: Counterfactual data augmentation using locally factored dynamics. Advances in Neural Information Processing Systems **33**, 3976–3990 (2020)
11. Richter, M.M., Weber, R.O.: Case-Based Reasoning: A Textbook. Springer, Berlin, Heidelberg (2013)
12. Rodriguez-Almeida, A.J., Fabelo, H., Ortega, S., Deniz, A., Balea-Fernandez, F.J., Quevedo, E., Soguero-Ruiz, C., Wägner, A.M., Callico, G.M.: Synthetic patient data generation and evaluation in disease prediction using small and imbalanced datasets. IEEE Journal of Biomedical and Health Informatics **27**(6), 2670–2680 (2023)
13. Smyth, B., Keane, M.T.: A few good counterfactuals: generating interpretable, plausible and diverse counterfactual explanations. In: International Conference on Case-Based Reasoning, pp. 18–32. Springer (2022)
14. Temraz, M., Keane, M.T.: Solving the class imbalance problem using a counterfactual method for data augmentation. Machine Learning with Applications **9**, 100,375 (2022)

15. Ustun, B., Spangher, A., Liu, Y.: Actionable recourse in linear classification. In: Proceedings of the conference on fairness, accountability, and transparency, pp. 10–19 (2019)
16. Verma, S., Dickerson, J., Hines, K.: Counterfactual explanations for machine learning: Challenges revisited. arXiv preprint arXiv:2106.07756 (2021)
17. Wachter, S., Mittelstadt, B., Russell, C.: Counterfactual explanations without opening the black box: Automated decisions and the gdpr. Harv. JL & Tech. **31**, 841 (2017)
18. Warren, G., Smyth, B., Keane, M.T.: "better" counterfactuals, ones people can understand: psychologically-plausible case-based counterfactuals using categorical features for explainable ai (xai). In: International conference on case-based reasoning, pp. 63–78. Springer (2022)
19. Weber, R., Shrestha, M., Johs, A.J.: Knowledge-based xai through cbr: There is more to explanations than models can tell. arXiv preprint arXiv:2108.10363 (2021)
20. Wiratunga, N., Wijekoon, A., Nkisi-Orji, I., Martin, K., Palihawadana, C., Corsar, D.: Discern: Discovering counterfactual explanations using relevance features from neighbourhoods. In: 2021 IEEE 33rd International Conference on Tools with Artificial Intelligence (ICTAI), pp. 1466–1473. IEEE (2021)
21. Yang, W., Li, J., Xiong, C., Hoi, S.C.H.: Mace: An efficient model-agnostic framework for counterfactual explanation. arXiv preprint arXiv:2205.15540 (2022)