Robustness to Spurious Correlations in Text Classification via Automatically Generated Counterfactuals

Zhao Wang[1] and **Aron Culotta** [2]

[1] Department of Computer Science, Illinois Institute of Technology, Chicago, IL
[2] Department of Computer Science, Tulane University, New Orleans, LA
zwang185@hawk.iit.edu, aculotta@tulane.edu

Abstract

Spurious correlations threaten the validity of statistical classifiers. While model accuracy may appear high when the test data is from the same distribution as the training data, it can quickly degrade when the test distribution changes. For example, it has been shown that classifiers perform poorly when humans make minor modifications to change the label of an example. One solution to increase model reliability and generalizability is to identify causal associations between features and classes. In this paper, we propose to train a robust text classifier by augmenting the training data with automatically generated counterfactual data. We first identify likely causal features using a statistical matching approach. Next, we generate counterfactual samples for the original training data by substituting causal features with their antonyms and then assigning opposite labels to the counterfactual samples. Finally, we combine the original data and counterfactual data to train a robust classifier. Experiments on two classification tasks show that a traditional classifier trained on the original data does very poorly on human-generated counterfactual samples (e.g., 10%-37% drop in accuracy). However, the classifier trained on the combined data is more robust and performs well on both the original test data and the counterfactual test data (e.g., 12%-25% increase in accuracy compared with the traditional classifier). Detailed analysis shows that the robust classifier makes meaningful and trustworthy predictions by emphasizing causal features and de-emphasizing non-causal features.

Introduction

Despite the remarkable performance machine learning models have achieved in various tasks, studies have shown that statistical models typically learn correlational associations between features and classes, and model validity and reliability are threatened by spurious correlations. Examples include: a sentiment classifier learns that "Spielberg" is correlated with positive movie reviews (Wang and Culotta 2020); a toxicity classifier learns that "gay" is correlated with toxic comments (Wulczyn, Thain, and Dixon 2017); a medical system learns that the disease is associated with patient ID (Kaufman, Rosset, and Perlich 2011); an object detection system recognizes a sheep based on the grass in the background (Ghorbani, Abid, and Zou 2019). If these kinds of spurious correlations are built into the model during training time, the model could fail when test data has a different distribution or even on samples with minor changes, and the predictions will be problematic and suffer from algorithm fairness or trust issues.

One solution to achieve robustness is to learn causal associations between features and classes. E.g., in the sentence *"This was a free book that sounded boring to me"*, the word most responsible for the label being negative is "boring" instead of "free". Identifying causal associations provides a way to build more robust and generalizable models.

Recent works try to achieve robustness with the aid of human-in-the-loop systems. Srivastava, Hashimoto, and Liang (2020) present a framework to make models robust to spurious correlations by leveraging human common sense of causality. They augment training data with crowd-sourced annotations about reasoning of possible shifts in unmeasured variables and finally conduct robust optimization to control worst-case loss. Similarly, Kaushik, Hovy, and Lipton (2020) ask humans to revise documents with minimal edits to change the class label, then augment the original training data with the counterfactual samples. Results show that the robust classifier is less sensitive to spurious correlations. While these prior works show the potential of using human annotations to improve model robustness, collecting such annotations can be costly.

In this paper, we propose to train a robust classifier with automatically generated counterfactual samples. Specifically, we first identify likely causal features using the closest opposite matching approach and then generate counterfactual training samples by substituting causal features with their antonyms and assigning opposite labels to the newly generated samples. Finally, we combine the original training data with counterfactual data to train a more robust classifier.

We experiment with sentiment classification tasks on two datasets (IMDB movie reviews and Amazon kindle reviews). For each dataset, we have the original training data and testing data, and additional human-generated counterfactual testing data. We first train a traditional classifier using the original data, which performs poorly on the counterfactual testing data (i.e., 10%-37% drop in accuracy). Then, we train a robust classifier with the combination of orig-

inal training data and automatically-generated counterfactual training data, and it performs well on both the original testing data and the counterfactual testing data (i.e., 12% - 25% absolute improvement over the baseline). Additionally, we consider limited human supervision in the form of human-provided causal features, which we then use to generate counterfactual training samples. We find that a small number of causal features (e.g., 50) results in accuracy that is comparable to a model trained with $1.7K$ human-generated counterfactual training samples from previous work.

Related Work

Spurious correlations are problematic and could be introduced in many ways. Sagawa et al. (2020) investigate how overparameterization exacerbates spurious correlations. They compare overparameterized models with underparameterized models and show that overparameterization encodes spurious correlations that do not hold in worst-group data. Kiritchenko and Mohammad (2018) showed that training data imbalances can lead to unintended bias and unfair applications (e.g., bias towards gender, race). Besides that, data leakage (Roemmele, Bejan, and Gordon 2011) and distribution shift between training data and testing data (Quionero-Candela et al. 2009) are particularly challenging and hard to detect as they introduce spurious correlations during model training and hurt model performance when deployed. Another new type of threat is backdoor attack (Dai, Chen, and Li 2019), where an attacker intentionally poisons a model by injecting spurious correlations into training data and manipulating model performance by specific triggers.

A growing line of research explores the challenges and benefits of using causal inference to improve model robustness. Wood-Doughty, Shpitser, and Dredze (2018) uses text classifiers in causal analyses to address issues of missing data and measurement error. Keith, Jensen, and O'Connor (2020) introduce methods to remove confounding from causal estimates. Paul (2017) proposes a propensity score matching method to learn meaningful causal associations between variables. Jia et al. (2019) consider label preserving transformations to improve model robustness to adversarial perturbations with Interval Bound Propagation. Landeiro and Culotta (2018) address the issue of spurious correlations by doing back-door adjustment to control for known confounders. Wang and Culotta (2020) train a classifier to distinguish between spurious features and genuine features, and gradually remove spurious features to improve worst-case accuracy of minority groups.

Recent works investigate how additional human supervision can reduce spurious correlations and improve model robustness. Roemmele, Bejan, and Gordon (2011) and Sap et al. (2018) show that humans achieve high performance on commonsense causal reasoning and counterfactual tasks. Zaidan and Eisner (2008) ask annotators to provide rationales as hints to guide classifiers paying attention to relevant features. Lu et al. (2018) and Zmigrod et al. (2019) use counterfactual data augmentation to mitigate bias. Ribeiro et al. (2020) evaluate model robustness using generated counterfactuals that requires significant human intervention (either by specifying substitution terms or generating templates and labeling examples). Garg et al. (2019) presume a predefined set of 50 counterfactually fair tokens and augment the training data with counterfactuals to improve toxicity classifier fairness.

While recent works have proposed the idea of generating and augmenting with counterfactuals for robust classifications, the main contributions of this paper are as follows:

- We propose to discover likely causal features using statistical matching techniques.
- Using these features, we automatically generate counterfactual samples by substituting causal features with antonyms, which significantly reduces human effort.
- We conduct experiments demonstrating the improved robustness of the resulting classifier to spurious features.
- We conduct additional analyses to show how the robust classifier increases the importance of causal features and decreases the importance of spurious features.

Problem and Motivation

To train a classification model, we fit a function $f(\cdot)$ with a set of labeled data and learn a map between input features and output labels. We consider a binary text classification task with the simple approach of logistic regression model[1]: $f(x;\theta) = \frac{1}{1+e^{-\langle x,\theta\rangle}}$ using bag-of-words features. Specifically, each document is a sequence of words $d = \langle w_1 \dots w_k \rangle$ that is transformed into a feature vector x via one-hot representation $x = \langle x_1 \dots x_V \rangle$ (V is the vocabulary size), and has a binary label $y \in \{-1, 1\}$. The model is fit on a set of labeled documents $\mathcal{D} = \{(d_1, y_1) \dots (d_n, y_n)\}$, and parameters are estimated by minimizing the loss function $\mathcal{L}$: $\theta^* \leftarrow \arg\min_\theta \mathcal{L}(\mathcal{D}, \theta)$. We can examine the (partial) correlations between features and labels by model coefficients.

Spurious correlations are very common in statistical models and they could mislead classifiers. For example, in our experimental dataset of Amazon kindle reviews, the classifier learns that "free" has a strong correlation with negative sentiment because "free" has a high frequency in negative book reviews (e.g., "This was a *free* book that sounded boring to me"), and thus the classifier makes errors when predicting positive documents that contain "free".

Previous works have tried various methods to reduce spurious correlations (e.g., regularization, feature selection, back-door adjustment (Hoerl and Kennard 1970; Forman 2003; Landeiro and Culotta 2018)). However, a more direct solution is to learn meaningful causal associations between features and classes. While expressing causality in the context of text classification can be challenging, we follow the previous work (Paul 2017) to operationalize the definition of a causal feature as follows: term w is a **causal feature** in document d if, all else being equal, one would expect w to be a determining factor in assigning a label to d. For example, in the sentence "This was a free book that sounded *boring* to me", "boring" is primarily responsible for the negative sentiment. In contrast, the term "free" itself does not convey neg-

[1]Our approach is model agnostic. We focus on logistic regression for interpretability and clarity.

ative sentiment. We consider "boring" as a causal term and "free" as a non-causal term (*term* refers to *word feature*).

Our approach in this paper is to first identify such causal features and then use them to automatically generate counterfactual training samples. Specifically, for a sample (d, y), we get the corresponding **counterfactual** sample (d', y') by (i) substituting causal terms in d with their antonyms to get d', and (ii) assigning an opposite label y' to d'. Let's consider the previous example to see how augmenting with counterfactual samples might work. Traditional classifiers trained on original data learns that "free" is correlated with the negative class due to its high frequency in negative book reviews. For every negative document containing "free", we generate one corresponding counterfactual document. The counterfactual sample for "This was a free book that sounded *boring* to me"(neg) would be "This was a free book that sounded *interesting* to me"(pos). When augmenting the original training data with counterfactual data, "free" would get equal frequency in both classes for the ideal case (i.e, if we could generate counterfactual samples for all documents containing "free"). Thus, a classifier fit on the combined dataset should have a reduced coefficient for "free" and increased coefficients for "boring" and "interesting."

Methods

Our approach is a two-stage process: we first identify likely causal features and then generate counterfactual training data using causal features. To identify causal features, we consider the counterfactual framework of causal inference (Winship and Morgan 1999). If word w in document d were replaced with some other word w', how likely is it that the label y would change? Since conducting randomized control trials to answer this question is infeasible, we instead use matching methods (Imbens 2004; King and Nielsen 2019). The intuition is as follows: if w is a reliable piece of evidence to determine the label of d, we should be able to find a very similar document d' that (i) does not contain w, and (ii) has the opposite label of d. For example, (d, y) = ("This was a free book that sounded *boring* to me", neg) and (d', y') = ("This was a free book that sounded *interesting* to me", pos) would be an ideal match where substituting causal term "boring" with another term "interesting" flips the label. While this is not a *necessary* condition of a causal feature (there may not be a good match in a limited training set), in the experiments below we find this to be a fairly precise approach to generate a small number of high-quality causal features.

The full steps of our approach are as follows:

1. We first train an initial classifier and extract strongly correlated terms $\langle t_1 \dots t_k \rangle$ as candidate causal features. E.g., for logistic regression model, we would extract features with high magnitude coefficients. For more complex models, other transparency algorithms may be used (Martens and Provost 2014).
2. For each top term t and a set of documents containing t: $D_t = \langle d_1 \dots d_n \rangle$, we search for a set of matched documents $D'_t = \langle d'_1 \dots d'_n \rangle$ and get $D_{match} = \{(d_1, d'_1, score_1) \dots (d_n, d'_n, score'_n)\}$, where the score for each match is the context similarity of d_i and d'_i. The matched documents have *opposite* labels.
3. Then for each term t and its corresponding matching set D_{match}, we pick the tuple $(d_i, d'_i, score_i)$ that has the highest similarity score as the *closest opposite match*. We then identify likely causal features by picking those whose closest opposite matches have scores greater than a threshold (0.95 is used below).
4. We use PyDictionary[2] to get antonyms for causal terms.
5. For each training sample, we generate its counterfactual sample by substituting causal terms with antonyms and assigning an opposite label to the counterfactual sample.
6. Finally, we train a robust classifier using the combination of original training data and counterfactual data.

We provide more details on these steps below.

Identifying Likely Causal Features

We expect causal features to have at least some correlations with the target class, so we first fit an initial binary classifier $f(x; \theta)$ on original training data $\mathcal{D} = \{(d_1, y_1) \dots (d_n, y_n)\}$ and extract top terms $\langle t_1 \dots t_k \rangle$ that have relatively large magnitude coefficients (e.g., > 1 in experiments below).

For a top term t and a document d containing t, we let $d[\hat{t}]$ represent the context of removing t from d. We search for another document d' that (i) has $t' \in d'$ and $t \notin d'$, where t' is another top term, and (ii) d' has the opposite label with d. We use a best match approach to search for $d'[\hat{t'}]$ that has highest semantic similarity to $d[\hat{t}]$ among all possible $d^*[\hat{t^*}]$: $d' \leftarrow \arg\max_{d^*} \text{sim}(d[\hat{t}], d^*[\hat{t^*}])$. For a term t, we get a set of corresponding matches as $D_{match} = \{(d_1, d'_1, score_1) \dots (d_n, d'_n, score_n)\}$, where the score for each match is the semantic similarity between $d[\hat{t}]$ and $d'[\hat{t'}]$. Each context is represented by concatenating the last four layers of a pre-trained BERT model (i.e., recommended by (Devlin et al. 2019)). We then select the match $(d_i, d'_i, score_i)$ that has the highest score in D_{match} as the **closest opposite match** for t. Table 1 shows examples of the closest opposite matches.

From the previous step, we get the closest opposite match for each top term. We then identify terms with closest opposite match scores greater than 0.95 as likely causal terms.

To evaluate the quality of this approach, the left panel of Figure 1 shows terms annotated by a human as likely to be causal or not, plotted by both their closest opposite match scores as well as the magnitude of coefficients from the classifier trained on original data. We can see that terms with very high closest opposite match scores are very likely to be causal. Note that this is not necessarily the case for terms with high coefficients (y-axis). The high precision and low recall pattern is further supported by the right panel.

Selecting Antonyms for Causal Terms

After identifying causal terms, we search for their antonyms using PyDictionary. This package provides simple interfaces for getting meanings from WordNet and synonyms and

[2]https://github.com/geekpradd/PyDictionary

Original sentence	Matched sentence	Context similarity
This was an *amazing* book.	This was a *boring* book	0.977
It was a *boring* read.	The book was *great* and long.	0.998
This short story was a *disappointment*.	This was a *great* short story.	0.992
This is one of the *funniest* movies I have seen.	This is one of the *worst* movies I have ever seen.	0.980
Fantastic film.	*Terrible* film.	1.00

Table 1: Examples of Closest Opposite Matches with Corresponding Context Similarity Scores

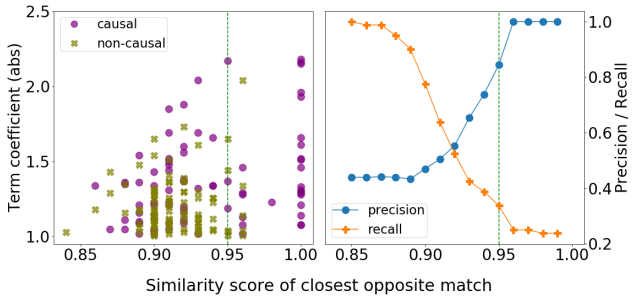

Figure 1: The "closest opposite match" score provides a high-precision indicator of causal features (IMDB dataset).

Causal Term	Antonyms
fantastic: 1.638	unimpressive: -0.462; inferior: -0.644
awesome: 1.202	unimpressive: -0.462
pleasant: 1.106	unpleasant: -0.333
dull: -1.881	lively: 0.302; colorful: 0.252
boring: -2.592	interesting: 0.734

Table 2: Discovered antonyms for causal terms and corresponding coefficients from the initial classifier.

antonyms from synonym.com. To reduce the noise of the returned antonyms, we require the antonyms to have opposite coefficients with the causal terms. Specifically, for each causal term t, we search for its antonyms by:

- First, check the direct antonyms for t and save those that satisfy the coefficient requirement as candidate antonyms.
- If no satisfying antonym is found, we then get synonyms for t and iteratively search for each synonym's antonyms, and save the satisfied antonyms as candidate antonyms.

After these two steps, we get at least one candidate antonym for each causal term $t : \{a_1 \dots a_k\}, k \geq 1$. Table 2 shows examples of the antonyms we get for causal terms.

Generating Counterfactual Samples

Next, for each training document d, we first identify all the causal terms in d: $\langle t_1 \dots t_m \rangle$, and then substitute all causal terms with their corresponding antonyms. If a causal term has multiple candidate antonyms, we randomly pick one to substitute. We only generate counterfactuals for documents contain at least one causal term. Finally, we assign opposite labels to the generated samples. Table 5 shows examples of generated counterfactual sentences. While most substitutions result in reasonable sentences, future work may investigate more sophisticated language models to ensure fluency of the generated counterfactuals.

Training a Robust Text Classifier

We augment the original training data with the automatically generated counterfactual data to train a robust classifier. We perform experiments below to investigate how do causal terms affect the quantity and quality of automatically generated counterfactual samples.

Data

We perform sentiment classification experiments on the following two datasets.[3] Each dataset has human-edited counterfactual testing samples to provide benchmark performance for classifier robustness.

IMDB movie reviews: This dataset is sampled from the original IMDB dataset (Pang and Lee 2005) and the counterfactual part is collected and published by Kaushik, Hovy, and Lipton (2020). They randomly sampled $2.5K$ reviews with balanced class distributions and partition them into 1707 training, 245 validation, and 488 testing samples. Then they instruct Amazon Mechanical Turk workers to revise each document with minimum changes towards a counterfactual label, and finally collected $2.5K$ counterfactually-manipulated samples.

Each document of this dataset is a long paragraph. We are both interested in exploring classifier performance for long texts and short texts. So, we additionally create a version of this dataset segmented into single sentences. To do so, we first fit a binary classifier on the original data and identify strongly correlated terms as keywords. Then we split each original document into single sentences and keep those containing at least one keyword. Sentence labels are inherited from the original document labels. To justify the validity of this approach, we randomly sampled 500 sentences and manually checked their labels. The inherited labels were correct for 484 sentences (i.e., 96.8% accuracy). We differentiate the IMDB dataset with long texts as **IMDB-L** and short texts as **IMDB-S**.

Amazon Kindle reviews (Kindle): This dataset contains book reviews from the Amazon Kindle Store and each review has a rating ranges from 1-5 (He and McAuley 2016). We label reviews with ratings $\{4,5\}$ as positive and reviews with ratings $\{1,2\}$ as negative, and then process this dataset to be single sentences following the approach used in IMDB.

[3]Code and data available at: https://github.com/tapilab/aaai-2021-counterfactuals

	IMDB-L		IMDB-S		Kindle	
	pos	*neg*	*pos*	*neg*	*pos*	*neg*
Train	856	851	4114	4059	5000	5000
Test	245	243	1144	1101	250	250
Top terms	231		198		194	
Causal terms	282		285		264	

Table 3: Dataset summary

Human edited counterfactuals: For the IMDB dataset, we have the human-generated counterfactual training data and counterfactual testing data. For kindle dataset, we randomly select 500 samples as test data (comparable size with the test data from IMDB-L) and manually edit them to be counterfactual samples with the minimum edits.

Ground truth causal terms: We manually annotated a set of ground truth causal terms for each dataset. Specifically, we asked two student annotators to label a term as causal if, all else being equal, this term is a determining factor in assigning a label to a document. While there is some subjectivity in the annotation, we did a round of training to resolve disagreements prior to annotation and the final agreement was generally high for this task (e.g., 96% raw agreement by fraction of labels that agree).

Table 3 shows the basic data statistics. For the top terms, we select them by thresholding on the magnitude of coefficients. For IMDB-L, we use threshold 0.4, and for IMDB-S and Kindle, we use threshold 1.0.

Experiments and Discussion

Causal Term Identification

According to the left panel of Figure 1, we find that the similarity scores of closest opposite matches seem to be a viable signal of true causal terms. The right panel shows the performance of identifying causal terms when thresholding on the closest opposite match scores. Using threshold 0.95, we identify 32 causal terms for IMDB-L and IMDB-S datasets, of which 27 are true causal terms (i.e., precision: 84%), and 23 causal terms for Kindle dataset, of which 19 are true causal terms (i.e., precision: 83%).

Robust Classification for Counterfactual Test Data

We fit five binary LogisticRegression classifiers with different training data (using scikit-learn (Pedregosa et al. 2011)) and evaluate their performance on the original test samples as well as counterfactual test samples. The training data compared below have increasing requirements for human supervision. For the first and second, only original training data is required. For the third and fourth, a human provides a list of causal terms, either by selecting from the list of top terms, or from the entire vocabulary. In the final setting, humans manually annotate counterfactual training samples (equivalent to the approach of Kaushik, Hovy, and Lipton (2020)). Details of the five levels of human supervision are as follows:

1. Only original training samples.
2. The original training samples are augmented with automatically generated counterfactual training samples using *predicted causal terms* .
3. The original training samples are augmented with counterfactual samples automatically generated using *human annotated causal terms from top words* (i.e., 65 for IMDB-L, 80 for IMDB-S, and 76 for Kindle).
4. The original training samples are augmented with counterfactual samples automatically generated using *human annotated causal terms from the entire vocabulary* (i.e., 282 for IMDB-L, 285 for IMDB-S, and 264 for Kindle).
5. The original training samples are augmented with *human-generated counterfactual* training samples.

We train the classifiers using the five different training sets and compare their performances on the original test samples and the human-generated counterfactual test samples. Table 4 shows the results.

When the classifier is trained on original training samples, it performs well on the original test data, but the accuracy degrades quickly when tested on human-generated counterfactual data (e.g., 20.1% absolute decrease for IMDB-L, 10.6% decrease for IMDB-S, 37.4% decrease for Kindle). This indicates that spurious correlations learned in the original classifier do not generalize well on the counterfactual test data.

When evaluating on human-generated counterfactual test samples, the classifier performance increases when we augment the original training data with counterfactual data. Even with no additional human supervision, the approach that automatically identifies causal terms outperforms the original classifier across all datasets (13%, 5.5%, 11% absolute improvement). Further improvements occur with additional human supervision in the form of causal terms. Using all causal terms (less than 300 terms per dataset), the approach achieves comparable performance to the more expensive baseline which requires humans to edit $> 1K$ counterfactual samples.[4]

We also observe that model accuracy slightly decreases on the original test data. This is because the spurious correlations hold in the original test data, but the importance of such features is reduced in the models trained on counterfactual samples. This suggests a potential tradeoff between accuracy on a specific dataset and generalizability of the model.

Alternative Experiments

The Appendix[5] provides additional results using more complex neural network models (LSTM with distributed word representations). The baseline classification accuracy is quite similar (within .03), and the relative accuracy of the different approaches exhibit very similar trends with the current results using logistic regression.

We have also run experiments to control for the training data size by downsampling the augmented training data to have the same size as original training data. Results show that there are only minor changes in accuracy (i.e., < 0.04),

[4]We lack human-generated counterfactual training samples for Kindle dataset, so we omit that result from Table 4.

[5]The Appendix is available in the Arxiv version of this paper.

Training data: *Original train samples + Counterfactual train samples*		*Testing data*					
		IMDB-L		**IMDB-S**		**Kindle**	
Counterfactual training samples	**Causal terms**	**Orig**	**CTF**	**Orig**	**CTF**	**Orig**	**CTF**
not used	not used	.816	.615	.711	.605	.888	.514
auto-generated	predicted from top words	.742	.744	.685	.660	.866	.624
	annotated from top words	.760	.818	.679	.696	.882	.662
	annotated from whole vocabulary	.773	**.857**	.685	**.726**	.752	**.720**
human-generated	not used	.818	**.869**	.705	**.762**	n/a	n/a

Table 4: Classification accuracy results. (CTF is human-generated counterfactual testing data.)

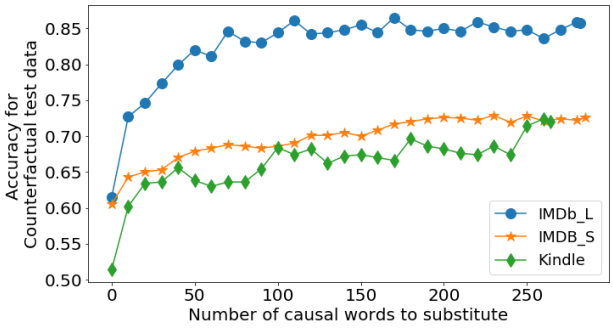

Figure 2: Performance change with counterfactuals generated using different numbers of annotated causal terms

and the overall trends match the current results (see Appendix).

Finally, as regularization terms may impact spurious features, we have also experimented with the L2 regularization term in logistic regression ($\{C = 0.01, 0.1, 1, 10, 100\}$), and there are only minor differences in accuracy on the counterfactual test data (see Appendix).

Performance Change with Different Number of Human Annotated Causal Terms

To further investigate how many human-provided causal terms are needed to improve robustness, Figure 2 shows the classification performance with different numbers of causal terms used for generating counterfactual samples. The quality of automatically generated counterfactuals depends on the causal terms used for antonym substitutions. We observe that performance seems to plateau after about 100 causal terms, which suggests that we can get similar performance by annotating 100 causal terms, as opposed to creating $> 1K$ counterfactual training samples. The cause of the plateau is likely due to the infrequency of subsequent terms and the fact that such terms co-occur with other causal terms, so they do not result in many new counterfactual samples.

Coefficient Change for Causal and Non-causal Terms

To understand why training with counterfactual data improves classifier robustness, Table 5 shows examples of the change in coefficients from the original classifier to the robust classifier for causal and non-causal terms.

Take the word "free" for example, which has a coefficient -1.41 in the original classifier due to its high frequency in negative samples. When generating counterfactuals, we substitute "boring" with "interesting" and generate a sample "This was a free book that sounded *interesting* to me" with positive sentiment. So the counterfactual samples increase the frequency of "free" in positive class and thus mitigate the spurious correlation of "free" with negative class. The classifier trained on a combination of original data and counterfactual data would learn a smaller magnitude coefficient for "free". Analogously, the approach increases the magnitude of causal terms such as "awesome" and "awful" by providing counterfactuals with opposite labels.

Error Analysis

Table 6 shows several test sentences that are misclassified by the original classifier and later corrected by the robust classifier. We can see again that the robust classifier increases coefficients of causal terms and decreases coefficients of non-causal terms. For example, "Really good movie" is incorrectly predicted as negative by the original classifier, because the causal term "good" has a small positive coefficient and the prediction is misled by the spuriously correlated negative term "movie"[6]. The robust classifier corrects this prediction by increasing the coefficient of the causal term "good" and decreasing the coefficient of the non-causal term "movie."

We conduct a final analysis to explore the impact of causal versus non-causal terms when correcting misclassifications. For each corrected sample, we compute separately the change in coefficient magnitudes for causal and non-causal terms. We then aggregate across all corrected samples to summarize the impact each type of correction has. As shown in Table 7, for IMDB-L, increasing coefficients of causal terms is more important than decreasing coefficients of non-causal terms, and the reverse is true for the other two datasets. This suggests that document length is an important factor in determining whether increasing coefficients of causal terms has bigger impacts or decreasing coefficients of non-causal terms has bigger impacts. Examining the average coefficient change of each term, the robust

[6]In the data, "film" correlates with high ratings, while "movie" correlates with low ratings.

	Term	Original coef	Robust coef	Original sentence	Counterfactual sentence
Non-causal terms	movie	-0.236	0.028	**Terrible** movie	**Fantastic** movie
	free	-1.41	-0.919	This was a free book that sounded **boring** to me.	This was a free book that sounded **interesting** to me.
Causal terms	awesome	0.584	1.838	He was an **awesome** actor.	He was an **awful** actor.
	terrible	-1.283	-2.336	The whole movie consists of **terrible** dialogue.	The whole movie consists of **pleasant** dialogue.

Table 5: Coefficient change of causal and non-causal terms.

Corrected samples	Original coef	Robust coef
Really good movie.(pos)	good:0.231 movie:-0.236	good:0.714 movie:0.028
The dubbing was as good as I have seen.(pos)	dubbing:-0.472 good:0.231	dubbing:-0.1 good:0.714
The story was incredibly interesting.(pos)	story:-0.171 incredibly:-0.874 interesting:-0.874	story:-0.083 incredibly:0.029 interesting:1.012

Table 6: Explanation for robust classifier corrected samples.

	Change per document		Change per term	
	causal	non-causal	causal	non-causal
IMDB-L	1.888	-0.734	0.327	-0.01
IMDB-S	0.435	-0.626	0.302	-0.042
Kindle	0.293	-0.772	0.315	-0.109

Table 7: Original versus robust classifier coefficient change for causal versus non-causal terms for corrected samples.

classifier tends to make bigger increases for causal terms and smaller decreases for non-causal terms. However, the greater frequency of non-causal terms can lead to these changes to have a greater overall impact on classification accuracy.

Conclusion and Future Work

We have presented a framework to automatically generate counterfactual training samples from causal terms and then train a robust classifier using the combination of original data and counterfactual data. Using this framework, we can easily improve classifier robustness even with few causal terms. If enough causal terms are annotated (e.g., 100 in our experiments), it is possible to achieve performance comparable to using human-generated counterfactuals. In future work, we will investigate extensions to increase the precision and recall of causal term identification to further reduce the reliance on human supervision. Additionally, it would be interesting to extend this framework to other tasks such as topic classification robustness. To do so, we would need to generalize the notion of "antonyms" to include terms that indicate a different topic (e.g., to convert a sports news story to a political news story, we might change the sentence "watch the game" to "watch the debate"). Then we could generate "counterfactuals" by substituting topic-related terms with terms that are not semantically related to the current topic (or related to other topics).

Acknowledgments

This research was funded in part by the National Science Foundation under grant #1618244. Zhao Wang was funded in part by a Dissertation Fellowship from the Computer Science department at Illinois Tech. We would also like to thank the anonymous reviewers for useful feedback.

References

Dai, J.; Chen, C.; and Li, Y. 2019. A Backdoor Attack Against LSTM-Based Text Classification Systems. *IEEE Access* 7: 138872–138878.

Devlin, J.; Chang, M.-W.; Lee, K.; and Toutanova, K. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *NAACL-HLT*.

Forman, G. 2003. An Extensive Empirical Study of Feature Selection Metrics for Text Classification. *J. Mach. Learn. Res.* 3: 1289–1305. ISSN 1532-4435.

Garg, S.; Perot, V.; Limtiaco, N.; Taly, A.; Chi, E. H.; and Beutel, A. 2019. Counterfactual Fairness in Text Classification through Robustness. In *Proceedings of the 2019 AAAI/ACM Conference on AI, Ethics, and Society*, AIES'19.

Ghorbani, A.; Abid, A.; and Zou, J. 2019. Interpretation of neural networks is fragile. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, 3681–3688.

He, R.; and McAuley, J. 2016. Ups and Downs. *Proceedings of the 25th International Conference on World Wide Web - WWW'16* doi:10.1145/2872427.2883037.

Hoerl, A. E.; and Kennard, R. W. 1970. Ridge Regression: Biased Estimation for Nonorthogonal Problems. *Technometrics* 12(1): 55–67. doi:10.1080/00401706.1970.10488634.

Imbens, G. W. 2004. Nonparametric estimation of average treatment effects under exogeneity: A review. *Review of Economics and statistics* 86(1): 4–29.

Jia, R.; Raghunathan, A.; Göksel, K.; and Liang, P. 2019. Certified Robustness to Adversarial Word Substitutions. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, 4129–4142. Hong Kong, China.

Kaufman, S.; Rosset, S.; and Perlich, C. 2011. Leakage in Data Mining: Formulation, Detection, and Avoidance. In *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '11, 556–563. New York, NY, USA. ISBN 9781450308137.

Kaushik, D.; Hovy, E.; and Lipton, Z. 2020. Learning The Difference That Makes A Difference With Counterfactually-Augmented Data. In *International Conference on Learning Representations*, ICLR'20.

Keith, K.; Jensen, D.; and O'Connor, B. 2020. Text and Causal Inference: A Review of Using Text to Remove Confounding from Causal Estimates. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, 5332–5344. Online: ACL.

King, G.; and Nielsen, R. 2019. Why propensity scores should not be used for matching. *Political Analysis* 27(4): 435–454.

Kiritchenko, S.; and Mohammad, S. 2018. Examining Gender and Race Bias in Two Hundred Sentiment Analysis Systems. In *Proceedings of the Seventh Joint Conference on Lexical and Computational Semantics*, 43–53. New Orleans, Louisiana: ACL. doi:10.18653/v1/S18-2005.

Landeiro, V.; and Culotta, A. 2018. Robust Text Classification under Confounding Shift. *Journal of Artificial Intelligence Research* 63: 391–419. doi:10.1613/jair.1.11248.

Lu, K.; Mardziel, P.; Wu, F.; Amancharla, P.; and Datta, A. 2018. Gender Bias in Neural Natural Language Processing. In *Logic, Language, and Security*, volume 12300, 189–202. Springer, Cham. doi:https://doi.org/10.1007/978-3-030-62077-6_14.

Martens, D.; and Provost, F. 2014. Explaining data-driven document classifications. *Mis Quarterly* 38(1): 73–100.

Pang, B.; and Lee, L. 2005. Seeing stars: Exploiting class relationships for sentiment categorization with respect to rating scales. In *Proceedings of the 43rd annual meeting on association for computational linguistics*, 115–124. ACL.

Paul, M. J. 2017. Feature Selection as Causal Inference: Experiments with Text Classification. In *Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017)*. Vancouver, Canada: ACL.

Pedregosa, F.; Varoquaux, G.; Gramfort, A.; Michel, V.; Thirion, B.; Grisel, O.; Blondel, M.; Prettenhofer, P.; Weiss, R.; Dubourg, V.; Vanderplas, J.; Passos, A.; Cournapeau, D.; Brucher, M.; Perrot, M.; and Duchesnay, E. 2011. Scikit-learn: Machine Learning in Python. *Journal of Machine Learning Research* 12: 2825–2830.

Quionero-Candela, J.; Sugiyama, M.; Schwaighofer, A.; and Lawrence, N. D. 2009. *Dataset Shift in Machine Learning*. The MIT Press. ISBN 0262170051.

Ribeiro, M. T.; Wu, T.; Guestrin, C.; and Singh, S. 2020. Beyond Accuracy: Behavioral Testing of NLP Models with CheckList. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. ACL.

Roemmele, M.; Bejan, C.; and Gordon, A. 2011. Choice of Plausible Alternatives: An Evaluation of Commonsense Causal Reasoning. In *AAAI Spring Symp - Technical Report*.

Sagawa, S.; Raghunathan, A.; Koh, P. W.; and Liang, P. 2020. An Investigation of Why Overparameterization Exacerbates Spurious Correlations. In *Proceedings of the 37th International Conference on Machine Learning,ICML 2020*.

Sap, M.; Bras, R. L.; Allaway, E.; Bhagavatula, C.; Lourie, N.; Rashkin, H.; Roof, B.; Smith, N. A.; and Choi, Y. 2018. ATOMIC: An Atlas of Machine Commonsense for If-Then Reasoning. *ArXiv* abs/1811.00146.

Srivastava, M.; Hashimoto, T.; and Liang, P. 2020. Robustness to Spurious Correlations via Human Annotations. In III, H. D.; and Singh, A., eds., *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, 9109–9119.

Wang, Z.; and Culotta, A. 2020. Identifying Spurious Correlations for Robust Text Classification. In *Findings of the Association for Computational Linguistics, EMNLP 2020*.

Winship, C.; and Morgan, S. L. 1999. The estimation of causal effects from observational data. *Annual review of sociology* 25(1): 659–706.

Wood-Doughty, Z.; Shpitser, I.; and Dredze, M. 2018. Challenges of Using Text Classifiers for Causal Inference. *Proceedings of the Conference on Empirical Methods in Natural Language Processing. EMNLP* 2018: 4586–4598.

Wulczyn, E.; Thain, N.; and Dixon, L. 2017. Ex Machina: Personal Attacks Seen at Scale. In *Proceedings of the 26th International Conference on World Wide Web*, WWW '17.

Zaidan, O. F.; and Eisner, J. 2008. Modeling Annotators: A Generative Approach to Learning from Annotator Rationales. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*, EMNLP '08. ACL.

Zmigrod, R.; Mielke, S. J.; Wallach, H.; and Cotterell, R. 2019. Counterfactual Data Augmentation for Mitigating Gender Stereotypes in Languages with Rich Morphology. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. ACL.