Explainable Sentiment Analysis using Counterfactuals and Shapley Values

Singapura Ravi Varun
varsi146@student.liu.se
732A81

A project presented for
Text Mining Course

STIMA
Linköping University
Sweden
12-01-2023

Abstract

This project is aimed at performing robust Sentiment Analysis by automatically and human generated counterfactuals and consequently explaining an ML model's predictions by using Shapley values(SHAP) to determine the influence of spurious patterns and whether it was eliminated. The in-domain datasets consists of original IMDB movie reviews(O), human revised counterfactuals(CF) and automatically generated counterfactuals(AC) variants of the original dataset. First, a sentiment classification is performed by using the SVM classifier by training and testing on different combinations of the variants of datasets mentioned above and then observing the performance metrics such as accuracy, precision and F1-score in each case. Then, open source LLMs such as BERT-base-uncased and ROBERTA are used to perform sentiment analysis by training and testing following a similar regime as mentioned and consequently the performance metrics are observed. Finally, an Explainable AI algorithm such as SHAP is used to visualize word/token contributions for predictions from SVM and BERT trained on various combination of training and test data in order to comment about the robustness of the models by eliminating influence of spurious patterns in the data.

1 Introduction

Recent advancements in deep neural networks such as the transformer architecture(Vaswani et al., 2017) based LLMs such as BERT(Devlin et al., 2019), ROBERTA(Liu et al., 2019) and other open and closed source models have shown that they outperform RNN based neural network architectures in all downstream tasks such as text classification, natural language inference, token classification and machine translation to name a few. The focus of this project however, is largely on Sentiment Analysis, a type of text classification task by using statistical models such as SVMs and LLMs such as BERT and ROBERTA.

However, if these ML models were to be implemented for Sentiment Analysis in critical and sensitive domains such as finance, law or medicine, then the onus is on the researchers and engineers to prove that the models are robust beyond reasonable doubt. Robust here means that these models have indeed captured the underlying patterns efficiently without learning spurious patterns and correlations. But, this is not the case as (Yang et al., 2021) and (Kaushik et al., 2021) have pointed out, and this is especially concerning in terms of generalization capacity on out of domain data(Yang et al., 2021). If these concerns were addressed in an assuring manner, there would be more wide-spread adaptation of these models in aforementioned sensitive domains.

Now that the problem associated with these models has been discussed, the remainder of this project will focus on applying Explainable AI paradigms such as counterfactual explanations, both human generated(Kaushik et al., 2021) and automatically generated(Yang et al., 2021) for building robust models in order to avoid learning spurious patterns. Then, the model predictions are interpreted and visualized using the SHAP(SHapley Additive exPlanations)(Lundberg and Lee, 2017) algorithm via a force plot of the sentence used for predicting its sentiment. The force plot is a visualization of the shapley values for each token/word in the sentence which is treated as a feature and helps interpreting which feature contributed the most in predicting the sentiment. By doing so, we compare the predictions in terms of the shapley values, of the model trained and tested on original data with models trained and tested on automatically generated and human generated counterfactuals and comment on the robustness induced due to the latter approach.

2 Theory

This section will deal with the brief introduction to the theoretical aspects of SVMs, Transformers followed by an in-depth explanation to Explainable AI concepts such as Counterfactuals and the Shapley values.

SVMs are a type of machine learning algorithm that falls in the supervised learning paradigm, where we have a set of labeled training data $(x_i, y_i), \ldots, (x_l, y_l)$ in $\mathbb{R}^k \times \mathbb{R}$ sampled according to some probability distribution $P(x, y)$ and the goal is to learn a model with a loss function $V(y, f(x))$ that can make accurate predictions on new, unseen data. The problem consists in finding a function f that minimizes the expectation of the error on new data, that is, find a function f that minimizes the expected error:(Evgeniou and Pontil, 2001)

$$\int V(y, f(x))P(x, y)\, dx\, dy$$

The transformer architecture which is the core of both BERT and ROBERTA, was a breakthrough since it is the first transduction model(Vaswani

et al., 2017) relying entirely on self-attention to draw global dependencies between input and output. This allows modelling of long input and output sequences without loss of dependencies or context. The authors define attention as "Scaled Dot-Product Attention", where the input consists of keys of dimension d_k, and values of dimension d_v and consequently compute dot products of queries with all keys, then divide each by $\sqrt{d_k}$, apply a softmax function to obtain the weights on the values.(Vaswani et al., 2017) In practice, the attention function is computed on queries simultaneously by packing them into a matrix Q, values and keys into V and K respectively:

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V$$

Now that a brief introduction to the models employed in this project has been provided, an in-depth theoretical discussion regarding Counterfactuals and SHAP follows. First, a formal definition for explainability/interpretability is required, one such definition is, **Interpretability is the degree to which a human can consistently predict the model's result.**(Kim et al., 2016) Interpretability and robustness of models is of utmost importance when these models are significantly impacting humans and is a useful debugging tool for detecting bias in ML models (Molnar, 2022). Interpretable methods are broadly classified as "Model-specific", "Model-Agnostic", Local and Global. Counterfactuals and the Shapley values are both Model-Agnostic and Local interpretation methods, meaning that they can be applied on any machine learning model and are applied after the model has been trained i.e., post-hoc(Molnar, 2022), and Local refers to the explanation of an individual prediction and not the entire model behaviour.

Algorithm 1 Generating plausible counterfactual instances

Input:Test document $D^{(n)} = \{P_1, P_2, \ldots, P_k\}$, with corresponding ground-truth labels Y, pre-trained Mask Language Model MLM, fine-tuned transformer classifier C, Positive Word Dictionaries POS, Negative Word Dictionaries NEG.
Output:Plausible counterfactual $D_{cf}^{(k)} = \{D_{rm}^{(n)}, D_{rep}^{(n)}\}$
for P_k in $D^{(n)}$ **do**
  for $S^{(i)}, Y_i$ in P_k **do**
    $\hat{S^{(i)}} \leftarrow \{w \in S^{(i)} \mid (w \in POS \wedge Y_i = pos) \vee (w \in NEG \wedge Y_i = neg)\}$
    $S_{sorted}^{(i)} \leftarrow sort(\hat{S^{(i)}}, key = \phi(w, \hat{S^{(i)}}))$ ▷ Eq.1
    $S_{rm}^{(i)} \leftarrow S_{sorted}^{(i)}[1:]$
    $S_{rep}^{(i)} \leftarrow S_{sorted}^{(i)}$
    for w in $S_{rep}^{(i)}$ **do**
      $W_p \leftarrow MLM(S_{mask(w)}^{(i)}, S_{rep}^{(i)})$
      $W_c \leftarrow \{w \in W_p \mid (w \in POS \wedge Y_i! = pos) \vee (w \in NEG \wedge Y_i! = neg)\}$
      $S_{rep}^{(i)}(w) \leftarrow sort(W_c, key = \phi(w, W_c))[0]$
    end for
    $P_{rm}^{(k)} \leftarrow P_{rm}^{(k)} + S_{rm}^{(i)}$
    $P_{rep}^{(k)} \leftarrow P_{rep}^{(k)} + S_{rep}^{(i)}$
  end for
  $D_{rm}^{(n)} \leftarrow D_{rm}^{(n)} + P_{rm}^{(k)}$
  $D_{rep}^{(n)} \leftarrow D_{rep}^{(n)} + P_{rep}^{(k)}$
end for
return $D_{rm}^{(n)}, D_{rep}^{(n)}$

where,

$$\phi(w, \hat{s}) = \mathbb{E}_{s_\beta}\left[\frac{l(s_{-\beta}; \hat{s}_\beta) - l(s_{-\beta}|p; \hat{s}_\beta)}{l(s_{-\beta}; \hat{s}_\beta)}\right] \quad (1)$$

The algorithm above(Yang et al., 2021) generates human-like counterfactual text instances by employing two key strategies: Removal (RM-CT) and Replacement (REP-CT). RM-CT identifies and removes key sentiment words from sentences, altering the classification outcome. REP-CT substitutes sentiment words with antonyms to craft a plausible alternative sentence(Yang et al., 2021). Both methods rely on a scoring function to assess word importance, utilizing a pre-trained BERT model for candidate selection. This process iterates over each sentence to produce two counterfactual documents, aiming to test and enhance sentiment classifier robustness.

A counterfactual is defined as a hypothetical that contradicts an observed fact(Molnar, 2022), for example the original dataset may consist of an entry *"I love running"*, then its corresponding counterfactual would be *"I dislike running"*. In the context of Sentiment Analysis, the original data point has a positive sentiment and the counterfactual has a negative sentiment, then, a counterfactual explanation can be described as the smallest change to the feature values(in this case, like to dislike) in order to flip the sentiment prediction. The authors(Kaushik et al., 2021) employ a human-in loop system to generate counterfactuals and (Yang et al., 2021) employ an algorithm(see Algorithm 1) that automatically generates counterfactuals, both aimed at improving the robustness of sentiment analysis models such as SVMs, BERT and ROBERTA.

By looking at an ML model predictions from a game theoretic perspective, a prediction can be explained by assuming that each feature value of an instance is a "player" and prediction is the payout. Then, **Shapley values** is a method from coalitional game theory which tells us how to fairly distribute the payout of the prediction amongst the features(Molnar, 2022). In the context of Sentiment Analysis, each token is a "player" and the sentiment is of course the prediction, then, Shapley values would help us in interpreting which token contributed the most towards the sentiment of an instance. Generally speaking, Shapley value is the average marginal contribution of a feature value across all possible coalitions.

Let us take for example, the sentence *"I love running"*, and we not only want to predict the sentiment but also understand how a model arrived at that prediction using Shapley Values. First, we would have to list all possible coalitions among the features/tokens and they are as follows:

- "I"
- "love"
- "running"
- "I", "love"
- "I", "running"
- "love", "running"
- "I", "love", "running"

The next step is to compute a sentiment score for each coalition of the features using a model. Then, for each feature, its respective marginal contribution to each possible coalition it can be a part of. For example, for the feature "love":

- Its contribution when it joins the feature "I" to form "I love"
- Its contribution when it joins the feature "running" to form "love running"
- Repeat the above step for each possible combination.

Finally, for each feature, average its marginal contributions across all possible combinations and the Shapley value for a word is this average which represents its fair share of contribution to the overall prediction.

The author (Molnar, 2022) defines the Shapley value via a value function val of players in S. Since the Shapley value of a feature is its contribution to the payout, weighted and summed over all possible feature value combinations, it is defined as:

$$\phi_j(\text{val}) = \sum_{\substack{S \subseteq \{1,\ldots,p\} \\ j \notin S}} \frac{|S|!(p-|S|-1)!}{p!} \times (\text{val}(S \cup \{j\}) - \text{val}(S))$$

where S is a subset of the features used in the model, x is the vector of feature values of the instance to be explained and p the number of features. $\text{val}_x(S)$ is the prediction for feature values in set S that are marginalized over features that are not included in set S:

$$\text{val}_x(S) = \int f(x_1, \ldots, x_p) d\mathbb{P}_{\{X \notin S\}} = \mathbb{E}_X[f(X)]$$

With this, the average marginal contributions of each feature can be computed and provide an insight into which feature contributed the most towards predicting the sentiment of an instance. Additionally, Shapley value has some desirable properties such as **Efficiency**(sum of all marginal contributions add up to the payout), **Symmetry**(two features with identical contribution to a coalition receive reward/payout), **Dummy**(any feature that does not change the predicted value, regardless of coalition should have a shapley value of 0) and **Additivity**(contribution value assigned to a feature should be the sum of the contributions from different models if the feature is part of multiple models), which together can be considered a definition of a fair payout(Molnar, 2022).

Model	Training/Test						
	O/O	O/C	O/AC	O/CF	CF/O	C/O	AC/O
SVM	85%	67%	63%	50%	53%	84%	80%
BERT	92%	93%	78%	93%	86%	92%	92%
RoBERTA	94%	94%	71%	93%	94%	94%	94%

Table 1: This table shows the accuracy of SVM, BERT and ROBERTA trained and tested on different combinations of datasets. The abbreviations of each of dataset are defined above in the report. "C" here refers to the combined dataset consisting of human generated counterfactuals and the original dataset.

3 Data

The data used in this project is the IMDB movie reviews which have already been annotated by the authors(Kaushik et al., 2021)[1] and is called as the original data(O). The authors then generate a human-in loop revised counterfactuals called as the human generated counterfactuals(CF), and the authors(Yang et al., 2021) generate counterfactuals automatically using their algorithm[2] to generate an **augmented** data set(AC) combining them with the original data. A pre-processing step before vectorizing the data using the TfidfVectorizer from sklearn and consequently using the SVM model to train on is to make sure the data(reviews) are lowercase, removing the stop words, retaining only alphabetical characters and finally lemmatizing the tokens.

4 Method

Since the aim of this project is to create robust models for Sentiment Analysis by the use of counterfactually augmented datasets and consequently using the SHAP python package(Lundberg and Lee, 2017) to interpret results. First, the performance of a simple statistical model such as an SVM classifier is evaluated by training it on "O" dataset with 1707 samples and test it on a hold-out dataset consisting of 488 samples and then the performance of this model against the "CF" and "AC" testing datasets is evaluated and a severe drop in accuracy of the classifier is observed. Next, the classifier is trained on the training datasets of CF and AC separately and its performance on the testing dataset of "O" is evaluated, and observe an increase in accuracy and by doing so it provides evidence that the influence of spurious patterns has been eliminated which was present in the former case(Yang et al., 2021). A similar training and testing procedure is followed by using pre-trained LLMs such as BERT and ROBERTA and the elimination of spurious patterns is observed by an increase in accuracy.

Now that the robust and non-robust models have been created, the SHAP package is used to visualize a force plot, summary plots and text plots of some predicted instances of reviews and we observe the contribution of features that led to a particular sentiment in both the cases. The visualizations are then used to compare the predictions and contributions of features/tokens in the robust and non-robust models and using the shapley values to further confirm that by including counterfactuals in the training sets the models are indeed robust. In other words, this can be thought of as a bridge between robustness induced by counterfactuals and Shapley values. The github link to the relevant repository containing the code following the above methodology can be found here[3].

5 Results and Discussion

This section will present the results of model evaluation after following the methodology specified in the previous section. In terms of the influence of spurious patterns, it is evident from Table 1 that the accuracy of the linear model(SVM Classifier) trained on original data and tested on the hold-out data(also original, see O/O) is 85% but however, when it is trained on original data and tested on the hold-out data for human-generated counterfactuals, automatically generated counterfactuals and the challenge data(see O/CF, O/AC, O/C) there is a significant drop in accuracy, 50%, 63% and 67% respectively. This implies that the model performance has been severely affected due to learning of spurious patterns(Yang et al., 2021).

A similar pattern is observed in BERT and RoBERTA when comparing model accuracies in

[1] https://github.com/acmi-lab/counterfactually-augmented-data

[2] https://github.com/lijiazheng99/Counterfactuals-for-Sentiment-Analysis

[3] https://github.com/VarunRavi95/Text_Mining_Project

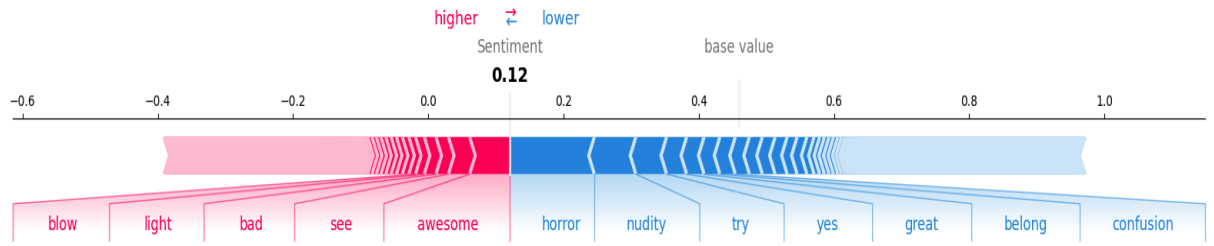

Figure 1: This is a force-plot of a misclassified example instance predicted by SVM.

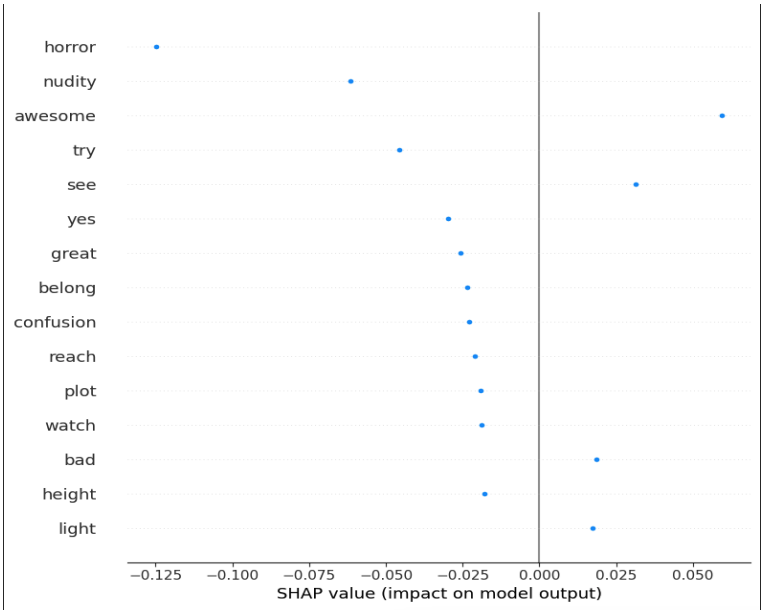

Figure 2: This is a summary-plot of a misclassified example instance predicted by SVM.

O/O and O/AC albeit not as drastic as the SVM model. In addition to this, on observing the model accuracies when trained on the augmented data and tested on the hold-out set of original data there is an increase of 14% in performance suggesting that the model has eliminated the effect of learning spurious patterns due to the augmentation of the original data with automatically generated counterfactuals. However, it is noteworthy that the model accuracies in O/C and O/CF are not affected and this can be due to two reasons, one being that training and test sets of the original, challenge and human generated counterfactuals are generated by the same group of labelers(Kaushik et al., 2021), the second reason is that the results suggest that large pre-trained Transformer models may be less sensitive to spurious patterns(Yang et al., 2021).

To further investigate the effect of spurious correlations and their drastic effect on the performance on the SVM classifier as mentioned above, we can produce a force-plot of prediction of an instance from the challenge data on a model trained on the original data. To do so, one instance that has been mis-classified by the SVM classifier will be considered, and movie review is as follows:

'i have seen many a horror flick in my time, some of them really awesome, but none reach the heights this film did. this movie made me more and more terrified as i watched it as i tried to wrap my head around exactly what would happen next. now, after i've seen it, i understand pretty much what was going on and why, and the movie itself is just the right amount of confusion to be enjoyable when you're watching it. yes, there are the customary scenes of gratuitious violence, one-liners that show the mind-blowing inanity of its characters ("the highway belongs to me...me!"), and enough nudity to sufficiently distract from the plot, but still you'll

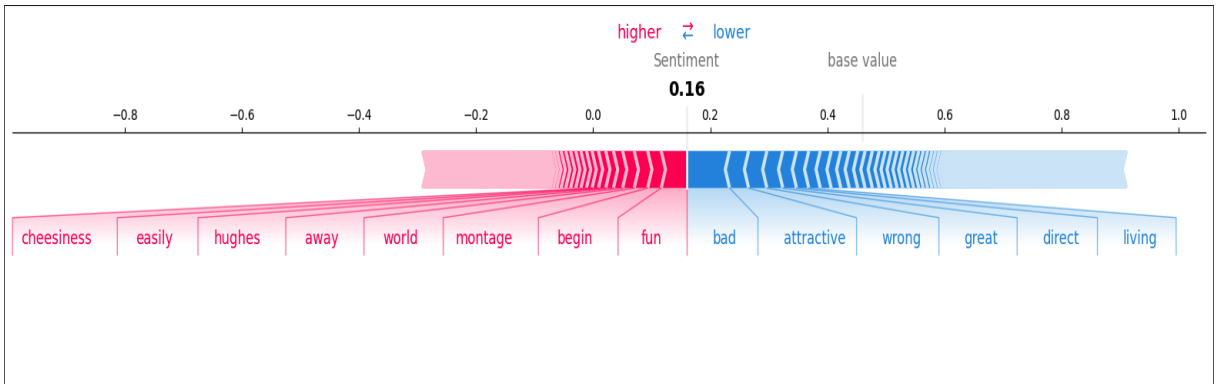

Figure 3: This is a force-plot of a correctly classified example instance.

leave this movie feeling happy to have seen it, and you'll be scared enough to sleep with the lights on.'

This is a positive review, however, the model has classified it as a negative review, and why the model did so can be interpreted using a force-plot. A force-plot is a depiction of each feature value that either increases or decreases a prediction. A prediction begins from the baseline and it is the average of all predictions, the arrows in the plot are representing shapley values that either pushes to increase or decrease a prediction, in this case, the sentiment score(Molnar, 2022). If the predicted sentiment score less than the base value, then it is a negative sentiment and vice-versa.

The Figure 1 shows features with the most contribution(shapley value) to the prediction of the sentiment, in this case, the red features/tokens drive the sentiment score higher and the blue features drive the score lower. On inspecting the top features(in terms of contribution), amongst the red features driving the sentiment towards a positive one, only the token "awesome" logically makes sense, while the rest of them do not contribute anything positive to the sentiment and this provides more evidence regarding the spurious patterns being learnt by the SVM model. On the other hand, amongst the blue features, while the features "horror", "nudity" and "none" have negative connotations in general, but in the context of the example instance, none of them have been used in a negative context.

The Figure 2 provides a insight into the relationship between the value of a feature and the impact on the prediction. The y-axis corresponds to the top features that contributed to the prediction and the x-axis represents the shapley value in terms of impact on prediction(Molnar, 2022) i.e., it may be a positive or negative impact represented by its distance from 0 on the x-axis. It provides more evidence that spurious patterns has been learnt since the features contributing positively towards the sentiment are not used in a positive context in the review except for the feature "awesome". Additionally, it also suggests that the word "horror" and "nudity" contributes in driving the sentiment towards the negative side in spite of its context in the review.

From the same model, considering an instance that was classified correctly, the review is as follows:

*'"bend it like beckham" reminds me of the worst of those 80ś teeny-bopper movies directed by john hughes. everything takes place in a bubble-gum colored world where everyone is attractive, there are some easily-resolved conflicts that occasionally take away from the mostly happy proceedings, and vast amounts of plot are summarized by montages set to bouncy pop tunes. everything is wrong with this, however. "bend it like beckham" is an absolute dog turd from beginning to end. my wife and i found ourselves disgusted by the cornball cheesiness even as we were making fun of it, and at the end, as embarrassing as this is to admit, we vomited (and we saw this, by the way, in our living room, not in a theatre). watch this movie and gouge your eyes out.

grade: d-"'*

Even though this instance was correctly classified, by looking at the force-plot from Figure 3, we see that other than the token "bad" none of the other tokens were used to convey a negative sentiment providing more evidence towards the influence of spurious patterns. A similar pattern is observed in the tokens driving the sentiment score upwards, only "fun" has a positive connotation, but in the context of the review it was actually used in a negative fashion.

Now, focussing on an SVM model trained to eliminate the above illustrated spurious pattern, we

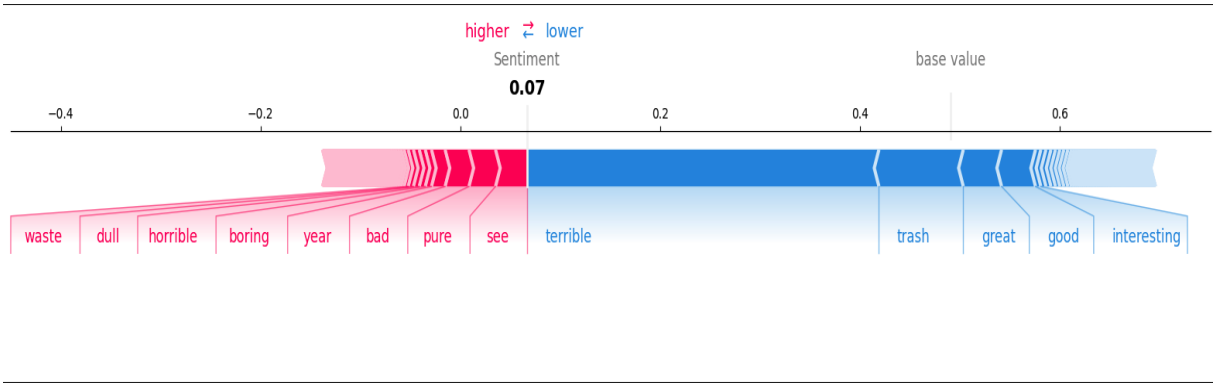

Figure 4: This is a force-plot of a correctly classified example instance from an SVM model trained to eliminate spurious patterns.

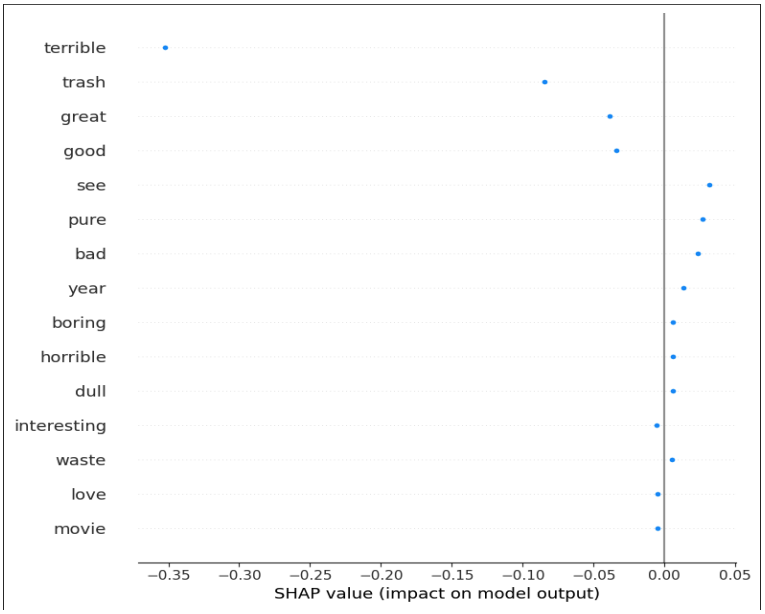

Figure 5: This is a summary-plot of a correctly classified example instance from an SVM model trained to eliminate spurious patterns.

choose a model trained on original dataset augmented by counterfactuals and then use SHAP to explain an instance that was correctly classified, the review is as follows:

'if you haven't seen this, it's terrible. it is pure trash. i saw this about 17 years ago, and i'm still screwed up from it.'

The review in ground truth has a negative sentiment and the model has also predicted a negative sentiment. From Figure 4, we can see that amongst the blue features, i.e., the ones decreasing the sentiment score the features "terrible" and "trash" are the ones with the most impact, which is as expected. Additionally, from Figure 5, we can see that the contribution of the featuers "terrible" and "trash" towards a negative sentiment are the highest and the rest are negligible. From these two plots, we can conclude that the model trained on the original dataset augmented with counterfactuals has indeed helped in eliminating the spurious patterns learnt by the model.

A similar investigation can be done to observe the influence of spurious patterns in BERT. We first consider a model trained on original dataset and predict an instance from the hold out dataset for original data and observe the effects. The following is the review:

"after catching some disease in space, an astronaut comes back to earth and starts melting.) this is the kind of movie that shouldn't have been made in the first place. "

The model predicts this as a negative sentiment, when we look at the text plot in Figure 6, the token "shouldn't" has the highest contribution in decreas-

Figure 6: This is a text plot of an instance predicted by BERT with spurious pattern.

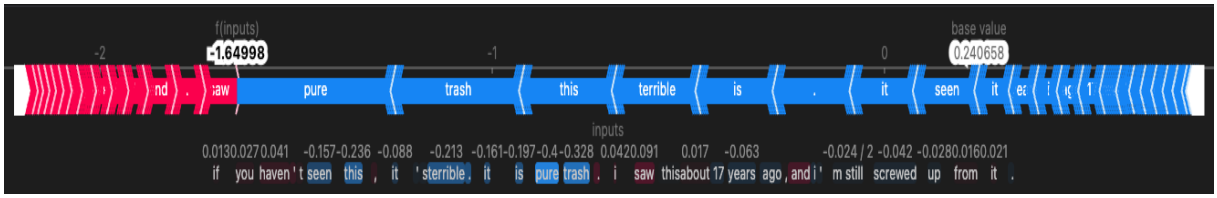

Figure 7: This is a text plot of an instance predicted by BERT trained to eliminate spurious patterns.

ing the sentiment score(-0.266), which is correct, but, the token ")" also has a similar contribution of -0.228, which provides more evidence for the effect of spurious pattern.

Now a model trained on augmented data, in order to eliminate the spurious correlations is chosen, and then we observe the contribution of the features. The review is as follows:

"if you haven't seen this, it's terrible. it is pure trash. i saw this about 17 years ago, and i'm still screwed up from it."

From Figure 7, it can be observed that the tokens contributing the most towards a negative sentiment are "pure trash" with combined score -0.728 and "terrible" with a score of -0.213, which is logically sound and it suggests that by training the model on the original dataset augmented with counterfactuals has helped eliminate the influence of spurious patterns.

There are, however, some limitations in this project. For instance, a more statistical oriented approach could be taken to investigate the effect of spurious correlations learnt by the model rather than investigating solely how features/tokens contribute to the prediction of a sentiment. Furthermore, this project aimed at building a bridge between robustness of a model trained on an augmented dataset and the SHAP values of a certain prediction, this may not be intuitive for all readers and may warrant more theoretical and mathematical backing. The focus on accuracy, precision, and F1-score as the primary performance metrics might overlook other important aspects like recall, specificity, or the model's ability to handle class imbalances. Finally, this project relied on automatically generated counterfactuals and one could argue about the quality of the augmented dataset and how well they represent real-world scenarios or complex language nuances.

6 Conclusion

This project has successfully demonstrated the efficacy of using automatically and human generated counterfactuals and Shapley Values in enhancing the interpretability and robustness of sentiment analysis models. By integrating counterfactuals into the training data, we observed a significant improvement in model performance through experimentation with various data combinations and architectures such as SVM, BERT and RoBERTA, particularly in handling spurious patterns which often mislead traditional sentiment analysis approaches. The use of SHAP (SHapley Additive exPlanations) further contributed to a deeper understanding of model predictions, allowing us to discern the influence of individual features on the overall sentiment determination and provide insight into the elimination of influence of spurious patterns by augmenting the original dataset with counterfactuals. In doing so, it can be claimed that these methods help in building robust and transparent machine learning models for critical domains.

References

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding.

Theodoros Evgeniou and Massimiliano Pontil. 2001. Support vector machines: Theory and applications. volume 2049, pages 249–257.

Divyansh Kaushik, Amrith Setlur, Eduard Hovy, and Zachary C Lipton. 2021. Explaining the efficacy of counterfactually augmented data. *International Conference on Learning Representations (ICLR)*.

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

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

Scott M Lundberg and Su-In Lee. 2017. A unified approach to interpreting model predictions. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, pages 4765–4774. Curran Associates, Inc.

Christoph Molnar. 2022. *Interpretable Machine Learning*, 2 edition.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Linyi Yang, Jiazheng Li, Padraig Cunningham, Yue Zhang, Barry Smyth, and Ruihai Dong. 2021. Exploring the efficacy of automatically generated counterfactuals for sentiment analysis. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 306–316, Online. Association for Computational Linguistics.