Geometry matters: Exploring language examples at the decision boundary

Debajyoti Datta (dd3ar@virginia.edu)  
Shashwat Kumar  
Laura Barnes  
Tom Fletcher

1 Systems Engineering, University of Virginia, Charlottesville, Virginia, USA  
2 Computer Science Dept., University of Virginia, Charlottesville, Virginia, USA

Abstract

A growing body of evidence has suggested that metrics like accuracy overestimate the classifier’s generalization ability. Several state of the art NLP classifiers like BERT and LSTM rely on superficial cue words (e.g., if a movie review has the word “romantic”, the review tends to be positive), or unnecessary words (e.g., learning a proper noun to classify a movie as positive or negative). One approach to address this is how teachers discover gaps in a student’s understanding: by finding problems where small variations confuse the student, it’s possible to perturb and study NLP classifiers in a similar way. While several perturbation strategies like contrast sets or random word substitutions have been proposed, they are typically based on heuristics and/or require expensive human involvement. In this work, using tools from information geometry, we propose a principled way to quantify the fragility of an example for an NLP classifier. By discovering such difficult examples for several state of the art NLP models like BERT, LSTM, and CNN, we demonstrate their susceptibility to meaningless perturbations like noun/synonym substitution, causing their accuracy to drop down to 20 percent in some cases. Our approach is simple, architecture agnostic and can be used to study the fragilities of text classification models. All the code used will be made publicly available, including a tool to explore the difficult examples for other datasets.

1 INTRODUCTION

NLP classifiers have achieved state of the art performance in several tasks like sentiment analysis [Maas et al. 2011], semantic entailment [Bowman et al. 2015] and question answering [Rajpurkar et al. 2016]. Despite their successes, several studies have pointed out issues in features learnt by such classifiers. Gururangan et al. [2018] discovered that several high performing NLP models were using trivial features like vagueness and negation to perform classification. Geirhos et al. [2020] performed several experiments to discover that “romantic” movies tend to be classified as positive movie reviews due to the presence of unnecessary words like proper nouns, a phenomena they called shortcut learning. Even models like BERT [Devlin et al. 2018] relied on superficial cue words like “not” to infer the line of argumentation. Strategies like this Niven and Kao [2020] enable models to perform prediction without inherently using the semantic meaning of the sentence. Simple attributes like word lengths are also exploited by models for prediction [Poliak et al. 2018].

In order to address these issues, several perturbation based approaches have been proposed. Contrast Sets [Gardner et al. 2020] and Counterfactual examples [Kaushik et al. 2019] get human annotators to perform minimal token substitutions to change the classifier prediction. While useful in addressing biases, manual curation of datasets are often time consuming and require extensive efforts. Using unsupervised training to mitigate issues related to shortcut learning also has not been successful in practice [Niven and Kao 2020]. The problem has also been approached from an adversarial perspective, with several works focused on introducing at the character level [Gao et al. 2018], Ebrahimi et al. [2018] and/or through word substitutions [Li et al. 2016], Liang et al. [2018]. Due to the lack of coherence of the generated sentences in the adversarial examples future work in this area aimed at generated grammatically correct sentences such that they are indistinguishable from original dataset. In Alzantot et al. [2018], Garg and Ramakrishnan [2020], the authors use black-box based approach to generate semantically equivalent adversarial examples.

While interesting, most traditional formulations treat this input embedding space as flat, thus reasoning that the gradient of the likelihood in the input space gives us the direction
A neural network can be considered as a mapping $g$ between the sentence manifold $X$ on the left (represented as a Euclidean space for simplification) and the statistical manifold of probability distribution over outputs $Z$. The fisher metric defines a particular Riemannian metric over this manifold. Let us consider an $\epsilon$ ball around two sentences $x_1$ (blue circle) and $x_2$ (red circle). As we can see from the distortion of the blue circle, for the $x_1$, any location perturbation can only result in a small change over the models output probability distribution. The eigenvalue of the fisher matrix quantifies this local distortion. These examples are not useful from a fragility based augmentation point of view. Perturbing the red examples however can result in drastic changes in the models output probability distribution. By measuring this distortion using the fisher matrix locally, we can study which examples affect the model output more. B: We train a neural network to learn a decision boundary (black) to separate two gaussians and color each data point by the local distortion. As we can see from the colors, perturbing a point close to the decision boundary results in a larger change over the models output probability distribution than a point away from the boundary (blue circle).

which causes the most significant change in terms of likelihood. If we were however, to consider the discrete likelihood of the class probabilities as the model output and the input as a pullback of this output, the space of output probability distributions is certainly non-linear, and the euclidean distance metric no longer suffices. We show this schematically in figure 1A, where the transformation $g$ maps elements of the sentence manifold $X$ to a non-linear statistical manifold $Z$. A natural distance metric to consider on this manifold is the Fisher Information Metric which along with being a Hessian of the KL divergence, it also a useful distance measure between probability distributions. Furthermore, the Fisher Information Metric is invariant to transformations like changing the model architecture, provided the likelihood remains the same. We thus use the eigenvalues of the Fisher matrix to discover high fragility regions in the statistical manifold. In regions with high $\lambda_{max}$ small perturbations can cause large changes in the output probability distribution (Fig 1A, red circle). Linguistically, this corresponds to the classifier being susceptible to meaningless perturbations like noun, synonym substitutions. Similarly, in regions with low $\lambda_{max}$ no local perturbation can affect the classifier, resulting in the classifier to being resilient upto 20% percent of word substitutions.

Our main contribution can thus be summarized as the following. 1. We propose a second order statistic, the log of the largest eigen value of Fisher matrix in order to capture linguistic fragilities. 2. We extensively establish the empirical relationship between $\lambda_{max}$ and success probability of random word substitutions.

To the best of our knowledge, this is the first work analyzing properties of the fisher metric to understand classifier fragility in NLP. The rest of the paper is organized as follows: In Section 2, we summarize related work. In Section 3, we discuss our approach of computing the FIM and the gradient-based perturbation strategy. In Section 4, we discuss the results of the eigenvalues of FIM in synthetic data and sentiment analysis datasets with BERT and CNN. We also do extensive quantitative evaluations using 4 other text datasets. Finally, in Section 5, we discuss the implications of studying the eigenvalues of FIM for evaluating NLP models.

## 2 RELATED WORK

In NLP, machine learning models for classification rely on spurious statistical patterns of the text and use shortcut for learning to classify. These can range from annotation artifacts, as was shown by Goyal et al. [2017], Kaushik and Lipton [2018], Gururangan et al. [2018], spelling mistakes as in McCoy et al. [2019], or new test conditions that require world knowledge Glockner et al. [2018]. Simple decision rules that the model relies on are hard to quantify. Trivial patterns like relying on the answer “2” for answering questions of the format “how many” for the visual question answering dataset Antol et al. [2015], would correctly answer 39% of the questions Jia and Liang [2017] showed that adversarially inserted sentences that did not change the correct
Table 1: CNN, Counterfactual Examples: In difficult examples (larger eigenvalue), individual synonym/antonym substitutions are effective in changing the classifier output. In easy examples (smaller eigenvalue) multiple antonym substitutions simultaneously have no effect on the classifier output.

| Perturbed sentiment | Word substitutions |
|--------------------|--------------------|
| Positive difficult example ($\lambda = 4.58$) | This movie was on TV last night. I was very impressed. The movie is well acted and directed. | |
| Positive easy example ($\lambda = 0.008$) | I learned by watching a lot of BBC shows since childhood. | |

Table 2: CNN, IMDB dataset: Unlike the difficult examples (larger eigenvalue), word substitutions are ineffective in changing the classifier output for the easier examples (smaller eigenvalue). In difficult examples synonym or change of name, changes classifier label. In easy examples, despite multiple simultaneous antonym substitutions, the classifier sentiment does not change.

| Perturbed sentiment | Word substitutions |
|--------------------|--------------------|
| Positive difficult example ($\lambda = 5.25$) | I read the book, and I thought it was a good book. Simon Pegg plays a rather childish character. He seems to be a man of style and charm. | |
| Positive easy example ($\lambda = 0.008$) | I learned by watching a lot of BBC shows since childhood. | |

answer, would cause state of the art models to regress in performance in the SQUAD [Rajpurkar et al. 2016] question-answering dataset. [Glockner et al. 2018] showed that template-based modifications by swapping just one word from the training set to create a test set highlighted models’ failure to capture many simple inferences. [Dixon et al. 2018] evaluated text classifiers using a synthetic test set to understand unintended biases and statistical patterns. Using a standard set of demographic identity terms, the authors reduce the unintended bias without hurting the model performance. [Shen et al.] showed that word substitution strategies include stylistic variations that change the sentiment analysis algorithms for similar word pairs. Evaluations of these models through perturbations of the input sentence are crucial to evaluating the robustness of models.

Another issue of language recently has been that static benchmarks like GLUE [Wang et al. 2018] tend to saturate quickly because of the availability of ever-increasing compute and harder benchmarks are needed like SuperGlue by [Wang et al. 2019]. A more sustainable approach to this is the development of moving benchmarks, and one notable initiative in this area is the Adversarial NLI by [Nie et al. 2019], but most of the research community hardly validate their approach against this sort of moving benchmark. In the Adversarial NLI dataset, the authors propose an iterative, adversarial human-and-model-in-the-loop solution for Natural Language Understanding dataset collection, where the goal post continuously shifts about useful benchmarks and makes models robust by training the model iteratively on difficult examples. Approaches like never-ending learning by [Mitchell et al. 2018] where models improve, and test sets get difficult over time is critical. A moving benchmark is necessary since we know that improving performance on a constant test set may not generalize to newly collected datasets under the same condition [Recht et al. 2019], [Beery et al. 2018]. Therefore, it is essential to find difficult examples in a more disciplined way. Approaches based on geometry have recently started gaining traction in computer vision literature. [Zhao et al. 2019] et al used a similar approach for understanding adversarial examples in images, a formulation we extend to language in our work.

3 METHODS
Consider the manifold of all possible sentence embeddings $X$. We show this schematically in Figure 1A, left. Although, we represent this as a euclidean space for simplicity, in the general setting this manifold could be non linear. Let $x \in X$ be a sentence on this manifold, and $y$ be the label vector corresponding to this sentence. The neural network $p(y|x)$ maps each sentence to a probability distribution over $y$. The set of all such probability distributions forms a statistical manifold $Z$ (Figure 1A, right). Depending on the properties of the neural network, an $\eta$ change in $x$ (red circle/blue circle) can result in a large change in $p(y|x)$. We seek to quantify this change by measuring the KL divergence between the two original and $\eta$ perturbed sentences.
We propose the following procedure to calculate $\lambda$.

We now perform a Taylor expansion of the first term on the right hand side:

$$
-\frac{1}{2} E_p(y|x) \nabla_x \log p(y|x) + \frac{1}{2} \eta^T \nabla^2_x \log p(y|x) \eta + ... + E_p(y|x) \log p(y|x)
$$

Since the expectation of score is zero and the first and last terms cancel out, we are left with:

$$
= -\frac{1}{2} \eta^T \nabla^2_x \log p(y|x) \eta
$$

By studying the eigenvalues of the fisher matrix, we can quantify the local distortion. As we see in Figure 1, the red sentence has a larger $\lambda_{max}$, resulting in a large change in the prob distribution. The blue sentence has a much smaller $\lambda_{max}$, meaning that perturbations around this sentence are less likely to affect $p(y|x)$ and thus the model accuracy.

After getting the eigenvalues of the FIM, we can use the largest eigenvalue $\lambda_{max}$ to quantify how fragile an example is to linguistic perturbation. These examples, thus, are also more confusing and more difficult for the model to classify. We propose the following procedure to calculate $\lambda_{max}$.

**Algorithm 1** Algorithm for estimating fragility of an example

**Input:** $x$: Sentence representation, $f$: Neural Network

**Output:** $\lambda_{max}$

1. $p = f(x)$
2. $J = \nabla_x \log p$
3. $p_c = \text{duplicate}(p, \text{dim}[0])$
4. $G = p_c J^T$
5. $(\lambda_3, v_3) = \text{eigendecomposition}(G)$
6. $\text{return max}(\lambda_3)$

Table 3: BERT: Difficult Examples change sentiment with a single word substituted. Easy examples, however, retain positive sentiment despite multiple substitutions of positive words with negative words.

| Perturbed sentiment | Word substitutions |
|---------------------|-------------------|
| Positive difficult example ($\lambda_{max} = 0.78$) | OK, I didn’t like the idea of this movie. I’m in the age demographic, and I kinda identify with some of the stories. Even the sometimes tacky and meaningful dialogue seems realistic, and in a different movie would have been forgivable @the #notin trying as hard as possible not to trash this movie like the others did, but it’s easy when the filmmakers were trying very hard @the #notin trying as hard as possible not to trash this movie like the others did, but it’s easy when the filmmakers were trying very hard @the #notin trying as hard as possible not to trash this movie like the others did, but it’s easy when the filmmakers were trying very hard |
| Positive easy example ($\lambda_{max} = 0.38$) | This is the best and most original show seen in years. The more I watch it the more I fall in love with it. It’s cast is excellent – especially [Blank] @the #notin looking for a show that is [incomplete] I personally love this show, but I think the casting was [incomplete] |

3.1 MODEL DESCRIPTIONS

We use multiple model architectures to understand the implications of the FIM.

- **Convolutional Neural Network (CNN):** We use the same paradigm [Kim 2014] for text classification using GloVe [Pennington et al. 2014] embeddings.
- **Long Short Term Memory Network (LSTM):** LSTM’s have been used extensively in Natural Language Processing for variety of tasks like text classification and language modeling [Schmidhuber 2015].
- **Bag of Tricks for Efficient Text Classification(Fasttext):** We used the Fasttext based model [Joulin et al. 2017] for classifying text and examining the quantitative aspects of Fisher Information Metric since the model is incredibly computational efficiency and strong performance across multiple datasets.

- **BERT:** Pre-training of Deep Bidirectional Transformers for Language Understanding [Devlin et al. 2018] Large scale pretrained language models have become extremely popular for data efficiency when trained on downstream tasks. However it has been shown recently, that on large datasets performance increase from fine-tuning is often within 1% of BERT. Thus we limited our analysis of using BERT for qualitative analysis for verification of it’s relationship with FIM since accuracy difference is minimal and it is computation intensive to calculate the eigenvalue of the FIM of BERT based fine tuned models. We instead used BERT embeddings in downstream tasks for qualitative evaluation of FIM. For the Fasttext, CNN and LSTM models we used early stopping with a patience of 5.

3.2 DATASETS

- **IMDB:** Large Movie Review Dataset [Maas et al. 2011]: IMDB dataset is the most common sentiment analysis dataset (Positive/Negative) for text classification collected from movie reviews on the IMDb platform. We do extensive quantitative and qualitative evaluation on the IMDb dataset across multiple architectures.
Figure 2: On YelpReviewPolarity we observe a correlation coefficient of -0.4 between minimum perturbation strength and the difficulty score. Similarly, we observe a correlation of 0.35 between the difficulty score and empirical success of random word substitutions. This suggests that fim eigenvalue captures perturbation sensitivity in both embedding space and word substitutions.

4 DISCUSSION AND RESULTS

We now explore both quantitatively and qualitatively how Fisher Information metric relates to metrics like accuracy and investigate it’s use as finding examples that are susceptible to perturbations.

4.1 FIM CAPTURES RESILIENCE TO LINGUISTIC PERTURBATIONS

In this section we explore relationship of FIM with linguistic perturbations (synonym/antonym substitutions, noun replacements), token level substitutions (random word choice, nearest neighbors in GloVE embeddings) and embedding perturbations.

4.1.1 \(\lambda_{\text{max}}\) is correlated with fragility to word substitutions

In order to investigate the relationship between \(\lambda_{\text{max}}\) and linguistic fragility, we perform the following experiment: We randomly sample 1000 examples from Yelp Review Polarity dataset. For each sampled example, we try a batch of word substitutions based on nearest neighbours in glove embedding space. In each substitution attempt, we try to flip between 10 percentage of words at a time, and calculate the percentage of successful flips which change the classifier prediction \(p_{\text{flip}}\). We also calculate the \(\lambda_{\text{max}}\) for these examples using the procedure outlined in Algorithm 1.

As we see from Figure 2 and Table 1, On YelpReviewPolarity, we observe \(r\) values of 0.3, 0.38 and 0.38 for CNN, FastText and LSTM respectively. Despite FIM being defined in a continuous space, we observe a strong linear relationship with success probability of token substitutions, which are discrete. This could be indicative of the fact that flipping a small of tokens in a large sentence is akin to an \(\epsilon\) perturbation in embedding space, which gets captured by \(\lambda_{\text{max}}\). More examples of token substitution and relationship of FIM with accuracy are in the Appendix.
Figure 3: Top row: Low FIM examples (easy examples) suffer from minimal accuracy loss when token substitutions are performed whereas accuracy on high FIM examples are significantly low. Bottom row: Easy examples require minimal perturbation to change classifier prediction. The x-axis represents the norm of perturbation. This suggests FIM eigenvalue captures perturbation sensitivity in both embedding space and token substitutions.

4.1.2 $\lambda_{\text{max}}$ captures strength of the minimal sufficient perturbation

We were also interested in investigating the relationship between $\lambda_{\text{max}}$ and norm of the smallest vector (oriented along the largest eigenvector $e_{\text{max}}$) which can cause the classifier to flip it’s prediction. We perform the following experiment: We randomly sample 500 examples YelpReview. For each of these examples, we calculate $\lambda_{\text{max}}$ and $e_{\text{max}}$ and perturb with a strength of $\eta$, where

$$x_{\text{flip}} = x_{\text{orgin}} + \eta \frac{e_{\text{max}}}{||e_{\text{max}}||}$$

By performing binary search on $\eta$, we can determine the minimum perturbation strength sufficient to flip the classifier prediction.

As evident from Figure 2 and Table 4, we obtain $r$ values of -0.41, -0.36 and -0.39 for CNN, FastText and LSTM. These correlations suggest that low $\lambda_{\text{max}}$ examples are much more resilient, requiring $\eta$ around X in order to flip them. High $\lambda_{\text{max}}$ examples on the other hand, get perturbed even by applying small perturbations up to norm of 0.1 strength of the eigenvector.

4.1.3 Qualitative exploration of fragilities

We also qualitatively explore the nature of linguistic fragilities for high $\lambda_{\text{max}}$ examples. We sample high and low $\lambda_{\text{max}}$ examples from the two tails of the $\lambda_{\text{max}}$ distributions. We then perform several token level substitutions manually: synonym, antonym substitutions, noun, and article substitutions. The tokens selected for substitutions were not random, but instead an Integrated Gradient was used to select the tokens [Sundararajan et al. 2017]. Integrated gradients assign importance score to words by the network and thus provide a more methodical approach to word substitutions than random word substitutions for qualitative evaluations.

As seen in Table 2 either replacing favorite with preferred or “Kirsten Dunst” with any of the listed actors/actresses suffices to change the classifier’s prediction. Note that, “Megan Fox’s” name appears in the same review in the previous sentence. Similar in Table 5 for counterfactual examples, it’s sufficient to replace “exciting” with either an antonym (“boring” or “uninteresting”) or a synonym (“extraordinary” or “exceptional”). For easy examples however, despite trying to replace four or more high attribution words simultaneously with antonyms, the predicted sentiment did not change. Substitutions include “good” to “bad”, “unfunny” to “funny”, “factually correct” to “factually incorrect”. Even though the passage included words like “boredom”, a word that is generally associated with a negative movie review, the model
did not assign it a high attribution score. Consequently, we did not try to substitute these words for testing robustness or fragility of word substitutions.

For our models, difficult examples have a mix of positive and negative words in a movie review. The models also struggled with examples of movies that selectively praise some attributes like acting (e.g., “Exceptional performance of the actors got me hooked to the movie from the beginning”) while simultaneously use negative phrases (e.g., “however the editing was horrible”). Difficult examples also have high token attributions associated with irrelevant words like “nuclear,” “get,” and “an.” Thus substituting one or two words in difficult examples change the predicted label of the classifier. Similarly, easier examples have clearly positive reviews (e.g., “Excellent direction, clever plot and gripping story”). Combining integrated gradients with high FIM examples can thus yield insights into the fragility of NLP models.

Several interesting observations emerge: CNN, Fasttext and LSTMs as a consequence of being trained on less amount of data is fragile to substitutions like Noun substitutions. BERT models on the other hand, because of their pretraining are robust to these sort of substitutions. High FIM BERT examples are more robust to meaningless changes, but single word substitutions still cause classifier prediction to change compared to low FIM examples.

4.2 HIGH \( \lambda_{\text{MAX}} \) EXAMPLES CAUSE SUBSTANTIAL DROP IN ACCURACY

In order to study the effect of \( \lambda_{\text{max}} \) on classification accuracy, we obtain examples with low FIM and high FIM \( \lambda_{\text{max}} \) examples and plot classifier accuracy as a function of number \( n \). As we see from Figure 3, low \( \lambda_{\text{max}} \) examples retain accuracies up to 90 percent even after the test set has been replaced with perturbed examples. In the high \( \lambda_{\text{max}} \) cases however, the accuracy drops below 20 percent as we increase the number of examples.

We also investigated the effect of perturbation strength along largest eigenvector \( \varepsilon_{\text{max}} \) on classifier accuracy. As we see from Figure 3, part B, low \( \lambda_{\text{max}} \) examples are significantly more robust to perturbations that high \( \lambda_{\text{max}} \) examples, where even minimal perturbations of the weight of 0.1 can cause the classifier prediction to change.

5 DISCUSSION

As evident from our experiments above, \( \lambda_{\text{max}} \) is correlated with susceptibility to linguistic perturbations. Furthermore, \( \lambda_{\text{max}} \) is directly correlated to the minimum perturbation needed to flip the classifier prediction. Finally, we show a stark difference in the response of low and high \( \lambda_{\text{max}} \) examples, with the high \( \lambda_{\text{max}} \) examples, perturbed examples being susceptible to meaningless perturbations (e.g., noun/synonym substitutions).

These experiments have several interesting ramifications: Firstly, it’s risky to over rely on accuracy while studying the generalizability of NLP classifiers. Most of the classifiers we used in our experiments attain really high accuracy on the test set despite failing simple sanity checks (e.g., invariance to synonym substitutions). It’s thus important to identify the examples which are most susceptible to perturbations and test their resilience. The fisher information provides us with a theoretically motivated framework to address this issue. By extracting the high \( \lambda_{\text{max}} \) examples and perturbing them by using glove based synonym substitutions, we can construct a rolling test set for our NLP classifiers. Furthermore, by mining only the high \( \lambda_{\text{max}} \) examples, FIM provides NLP practitioners with an interactive way to perturb and explore the flaws of their classifiers, something which would be extremely tedious to do on the entire dataset otherwise. While the focus of our work was not to generate adversarial examples, an alternate framing of this problem is to consider high \( \lambda_{\text{max}} \) sentences to be more susceptible to adversarial attacks whereas low \( \lambda_{\text{max}} \) sentences are generally robust to significant adversarial perturbations.

6 CONCLUSION

In this paper, we introduced a method to discover the highly fragile examples for an NLP classifier. Our method discovered in several state of the art classifiers like BERT, CNN and LSTM. Furthermore, our experiments shed some light on the links between geometry of NLP classifiers and their linguistic and embedding space perturbability. Our method provides NLP practitioners with both an automated an interactive human in the loop framework to better understand their models.

There are several interesting extensions. Firstly, it would be interesting to decode the optimal perturbation vector \( \varepsilon_{\text{max}} \) associated with \( \lambda_{\text{max}} \) as a sentence. We are developing several approaches based on finding token substitutions which
maximize this dot product in order to address this. It will also be interesting to study the link between purely geometrical properties like distance to the decision boundary and linguistic resilience. Finally, we are also interested in measuring the norm of token substitutions in embedding space to understand the relationship between the two.

References

Moustafa Alzantot, Yash Sharma, Ahmed Elgohary, Bo-Jhang Ho, Mani Srivastava, and Kai-Wei Chang. Generating natural language adversarial examples. In Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, pages 2890–2896, 2018.

Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C Lawrence Zitnick, and Devi Parikh. Vqa: Visual question answering. In Proceedings of the IEEE international conference on computer vision, pages 2425–2433, 2015.

Sara Beery, Grant Van Horn, and Pietro Perona. Recognition in Terra Incognita. pages 456–473, 2018. URL http://openaccess.thecvf.com/content_ECCV_2018/html/Beery_Recognition_in_Terra_ECCV_2018_paper.html

Samuel R Bowman, Gabor Angeli, Christopher Potts, and Christopher D Manning. A large annotated corpus for learning natural language inference. arXiv preprint arXiv:1508.05326, 2015.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. arXiv preprint arXiv:1810.04805, 2018.

Lucas Dixon, John Li, Jeffrey Sorensen, Nithum Thain, and Lucy Vasserman. Measuring and mitigating unintended bias in text classification. In Proceedings of the 2018 AAAI/ACM Conference on AI, Ethics, and Society, pages 67–73, 2018.

Javid Ebrahimi, Anyi Rao, Daniel Lowd, and Dejing Dou. Hotflip: White-box adversarial examples for text classification. In Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers), pages 31–36, 2018.

Ji Gao, Jack Lanchantin, Mary Lou Soffa, and Yanjun Qi. Black-box generation of adversarial text sequences to evade deep learning classifiers. In 2018 IEEE Security and Privacy Workshops (SPW), pages 50–56. IEEE, 2018.

Matt Gardner, Yoav Artzi, Victoria Basmova, Jonathan Berant, Ben Bogin, Sihao Chen, Pradeep Dasigi, Dheeru Dua, Yanai Elazar, Ananth Gottumukkala, Nitish Gupta, Hanna Hajishirzi, Gabriel Ilharco, Daniel Khashabi, Kevin Lin, Jiangming Liu, Nelson F. Liu, Phoebe Mcilraith, Qiang Ning, Sameer Singh, Noah A. Smith, Sanjay Subramanian, Reut Tsfarfaty, Eric Wallace, Alty Zhang, and Ben Zhou. Evaluating NLP Models via Contrast Sets. arXiv:2004.02709 [cs], April 2020. URL http://arxiv.org/abs/2004.02709 arXiv: 2004.02709.

Siddhant Garg and Goutham Ramakrishnan. Bae: Bert-based adversarial examples for text classification. In Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), pages 6174–6181, 2020.

Robert Geirhos, Jörn-Henrik Jacobsen, Claudio Michaelis, Richard Zemel, Wieland Brendel, Matthias Bethge, and Felix A. Wichmann. Shortcut Learning in Deep Neural Networks. arXiv:2004.07780 [cs, q-bio], April 2020. URL http://arxiv.org/abs/2004.07780 arXiv: 2004.07780.

Max Glockner, Vered Shwartz, and Yoav Goldberg. Breaking nli systems with sentences that require simple lexical inferences. arXiv preprint arXiv:1805.02266, 2018.

Yash Goyal, Tejas Khot, Douglas Summers-Stay, Dhruv Batra, and Devi Parikh. Making the v in vqa matter: Elevating the role of image understanding in visual question answering. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 6904–6913, 2017.

Suchin Gururangan, Swabha Swayamdipta, Omer Levy, Roy Schwartz, Samuel R. Bowman, and Noah A. Smith. Annotation Artifacts in Natural Language Inference Data. arXiv:1803.02324 [cs], April 2018. URL http://arxiv.org/abs/1803.02324 arXiv: 1803.02324.

Robin Jia and Percy Liang. Adversarial examples for evaluating reading comprehension systems. arXiv preprint arXiv:1707.07332, 2017.

Armand Joulin, Édouard Grave, Piotr Bojanowski, and Tomáš Mikolov. Bag of tricks for efficient text classification. In Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 2, Short Papers, pages 427–431, 2017.

Divyansh Kaushik and Zachary C Lipton. How much reading does reading comprehension require? a critical investigation of popular benchmarks. arXiv preprint arXiv:1808.04926, 2018.

Divyansh Kaushik, Eduard Hovy, and Zachary C Lipton. Learning the difference that makes a difference with counterfactually-augmented data. arXiv preprint arXiv:1909.12434, 2019.

Yoon Kim. Convolutional neural networks for sentence classification. arXiv preprint arXiv:1408.5882, 2014.
Jiwei Li, Will Monroe, and Dan Jurafsky. Understanding neural networks through representation erasure. *arXiv preprint arXiv:1612.08220*, 2016.

Bin Liang, Hongcheng Li, Miaoqiang Su, Pan Bian, Xirong Li, and Wenchang Shi. Deep text classification can be fooled. In *Proceedings of the 27th International Joint Conference on Artificial Intelligence*, pages 4208–4215, 2018.

Andrew Maas, Raymond E Daly, Peter T Pham, Dan Huang, Andrew Y Ng, and Christopher Potts. Learning word vectors for sentiment analysis. In *Proceedings of the 49th annual meeting of the association for computational linguistics: Human language technologies*, pages 142–150, 2011.

R Thomas McCoy, Ellie Pavlick, and Tal Linzen. Right for the wrong reasons: Diagnosing syntactic heuristics in natural language inference. *arXiv preprint arXiv:1902.1007*, 2019.

Yixin Nie, Adina Williams, Emily Dinan, Mohit Bansal, Jason Weston, and Douwe Kiela. Adversarial nli: A new benchmark for natural language understanding. *arXiv preprint arXiv:1910.14599*, 2019.

Timothy Niven and Hung Yu Kao. Probing neural network comprehension of natural language arguments. In *57th Annual Meeting of the Association for Computational Linguistics, ACL 2019*, pages 4658–4664. Association for Computational Linguistics (ACL), 2020.

Jeffrey Pennington, Richard Socher, and Christopher D Manning. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543, 2014.

Adam Poliak, Jason Naradowsky, Aparajita Haldar, Rachel Rudinger, and Benjamin Van Durme. Hypothesis Only Baselines in Natural Language Inference. In *Proceedings of the Seventh Joint Conference on Lexical and Computational Semantics*, pages 180–191, New Orleans, Louisiana, June 2018. Association for Computational Linguistics. doi: 10.18653/v1/S18-2023. URL [https://www.aclweb.org/anthology/S18-2023](https://www.aclweb.org/anthology/S18-2023)

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. Squad: 100,000+ questions for machine comprehension of text. *arXiv preprint arXiv:1606.05250*, 2016.

Benjamin Recht, Rebecca Roelofs, Ludwig Schmidt, and Vaishaal Shankar. Do imagenet classifiers generalize to imagenet? *arXiv preprint arXiv:1902.10811*, 2019.

Jürgen Schmidhuber. Deep learning in neural networks: An overview. *Neural networks*, 61:85–117, 2015.

Judy Hanwen Shen, Lauren Fratamico, Iyad Rahwan, and Alexander M Rush. Darling or babygirl? investigating stylistic bias in sentiment analysis.

Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic attribution for deep networks. In *International Conference on Machine Learning*, pages 3319–3328. PMLR, 2017.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R Bowman. Glue: A multi-task benchmark and analysis platform for natural language understanding. *arXiv preprint arXiv:1804.07461*, 2018.

Alex Wang, Yada Pruksachatkun, Nikita Nangia, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel Bowman. Superglue: A stickier benchmark for general-purpose language understanding systems. In *Advances in Neural Information Processing Systems*, pages 3261–3275, 2019.

Xiang Zhang, Junbo Jake Zhao, and Yann LeCun. Character-level convolutional networks for text classification. In *NIPS*, 2015.

Chenxiao Zhao, P Thomas Fletcher, Mixue Yu, Yaxin Peng, Guixu Zhang, and Chaomin Shen. The adversarial attack and detection under the fisher information metric. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 5869–5876, 2019.

### A APPENDIX
Figure 4: Left figure: The difficult examples on the FIM test set are challenging for the classifiers with accuracy between 1-5%. For the easy examples, (low FIM $\lambda_{max}$) the accuracy ranges between 60-100%. Right figure: We plot the classifier accuracy with respect to the l2 norm of the perturbation in embedding space. For difficult examples, the classifier accuracy drops below 5% at small perturbation strengths (l2 norm < 0.3)

Figure 5: a) Distribution of difference in largest eigenvalue of FIM of the original and the perturbed sentence in contrast set and counterfactual examples for BERT and CNN. With a mean near 0, these perturbations are not difficult for the model. Adhoc perturbations are thus not useful for evaluating model robustness.