Shortcut Learning of Large Language Models in Natural Language Understanding

Mengnan Du
New Jersey Institute of Technology
Newark, NJ, USA
mengnan.du@njit.edu

Fengxiang He
University of Edinburgh
Edinburgh, UK
F.He@ed.ac.uk

Na Zou
Texas A&M University
College Station, TX, USA
nzou1@tamu.edu

Dacheng Tao
The University of Sydney
Sydney, Australia
dacheng.tao@gmail.com

Xia Hu
Rice University
Houston, TX, USA
xia.hu@rice.edu

ABSTRACT

Large language models (LLMs) have achieved state-of-the-art performance on a series of natural language understanding tasks. However, these LLMs might rely on dataset bias and artifacts as shortcuts for prediction. This has significantly affected their generalizability and adversarial robustness. In this paper, we provide a review of recent developments that address the shortcut learning and robustness challenge of LLMs. We first introduce the concepts of shortcut learning of language models. We then introduce methods to identify shortcut learning behavior in language models, characterize the reasons for shortcut learning, as well as introduce mitigation solutions. Finally, we discuss key research challenges and potential research directions in order to advance the field of LLMs.

1 INTRODUCTION

Natural language understanding (NLU) is a subfield of artificial intelligence that requires computer software to comprehend input in the form of sentences. Representative NLU tasks include natural language inference (NLI), question answering (QA), reading comprehension, etc. Furthermore, NLU has a number of real-world applications, including Amazon Alexa, Siri, and Google Assistant. The major characteristic of NLU tasks is that they are difficult and typically require world knowledge and commonsense reasoning. Recently, large language models (LLMs), such as BERT [8], RoBERTa [20], T5 [30], GPT-3 [4], have been reported to achieve state-of-the-art performance in a series of high-level NLU tasks. The LLM performance has even been reported to be significantly higher than human performance. However, the superior performance has only been observed in the benchmark test data that have the same distribution as the training set. Recent studies indicate that these LLMs are not robust and that the models do not remain predictive when the distribution of inputs changes [9, 24, 47]. Specifically, these LLMs have low generalization performance when applied to out-of-distribution (OOD) test data and are also vulnerable to various types of adversarial attack. This leaves us wondering: *Why are these LLMs not robust? Have these LLMs mastered the high-level semantic understanding and reasoning that we expect of them?*

A major reason for the low robustness of LLMs is **shortcut learning**. The shortcut learning behavior has also been called other names in the literature, such as learning bias, superficial correlations, right for wrong reasons, Clever Hans effect[1], etc. The shortcut learning behavior has been observed for a series of NLU tasks. For example, recent empirical analysis indicates that the performance of BERT-like models for the NLI task could be mainly explained by relying on spurious statistical cues such as unigrams 'not', 'do', 'is' and bigrams 'will not' (see Figure 1 (b)) [13, 24]. Similarly, for the reading comprehension task, the models rely on the lexical matching of words between the question and the original passage, while ignoring the designed reading comprehension task [17]. The current standard approach to training LLM is to use empirical risk minimization (ERM) on NLU datasets that typically contain various types of artifacts and biases. As such, LLMs have learned to rely on dataset artifacts and biases and capture their spurious correlations with certain class labels as shortcuts for prediction. The shortcut learning behavior has significantly affected the robustness of LLMs (see Figure 1 (a)), thus attracting increasing attention from the NLP community to address this problem.

In this work, we offer a comprehensive review of the shortcut learning problem in language models with a focus on medium-sized LLMs those typically having less than a billion parameters. The main emphasis is on the prevalent pre-training and fine-tuning paradigm utilized in NLU tasks (see Figure 3). We cover the concept of shortcut learning and robustness challenges in Section 2, detection approaches in Section 3, characterization of the corresponding reasons in Section 4, and mitigation approaches in Section 5. We also provide a further discussion of future research directions and connections with other directions in Section 6. Beyond the standard fine-tuning paradigm, in Section 7, we briefly discuss the challenges of shortcut learning posed by the prompt-based paradigm, especially regarding the massive language models which possess over a billion parameters, such as GPT-3 and T5.

Conference'17, July 2017, Washington, DC, USA

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM…$15.00
https://doi.org/XXXXXXX.XXXXXXX

[1]The eponymous horse appeared to be capable of performing simple intellectual tasks, but actually relied on involuntary cues given by its handler.

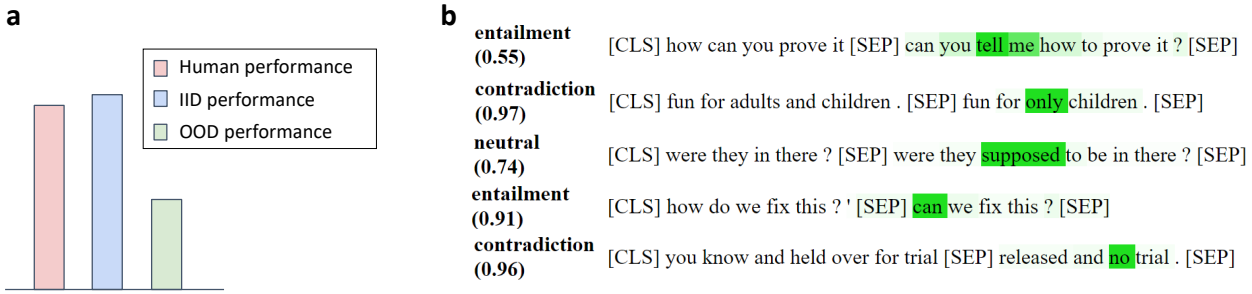

Figure 1: Shortcut learning behavior and its negative impact, taking natural language inference (NLI) task for example. The goal of NLI is to infer whether the relationship between two branches of input, i.e., premise and hypothesis, is entailment, contradiction, or neutral. (a) LLMs outperforms human performance for IID benchmark test set, while achieve much lower generalization performance on OOD test set. (b) A key reason is that LLMs primarily rely on the lexical bias and other kinds of shortcuts for prediction.

2 SHORTCUT LEARNING PHENOMENA

2.1 What is Shortcut Learning?

Features captured by the model can be broadly categorized as useless features, robust features, and non-robust features (see Figure 2). Shortcut learning refers to the phenomenon that LLMs (especially those trained with standard ERM-based method) highly rely on non-robust features as shortcuts, failing to learn robust features and capture high-level semantic understanding and reasoning. Non-robust features do help generalization for development and test sets that share the same distribution with training data. However, they cannot generalize to OOD test sets and are vulnerable to adversarial attacks. Non-robust features are oriented from biases in the training data and come in different formats. In the following, we introduce several representative ones.

- *Lexical Bias*: Some lexical features have a high correlation of co-occurrence with certain class labels. These lexical features mainly consist of low-level functional words such as stop words, numbers, negation words, etc. A typical example is the NLI task, where LLMs are highly dependent on unintended lexical features to make predictions [9, 24]. For example, these models tend to give contradiction predictions whenever there exist negation words in the input samples, e.g., 'never', 'no'.
- *Overlap Bias*: It occurs in NLU applications with two branches of text, e.g., natural language inference, question answering, and reading comprehension. LLMs use the overlap of features between the two branches of inputs as spurious correlations as shortcuts. For example, reading comprehension models use the overlap between the passage and the question pair for prediction rather than solving the underlying task [17]. Similarly, question answering models excel at test sets by relying on the heuristics of question and context overlap [35].
- *Position Bias*: The distribution of the answer positions may be highly skewed in the training set for some applications. The LLMs would predict answers based on spurious positional cues. Take the question answering task for example, the answers lie only in the k-th sentence of each passage [16]. As a result, question answering models rely on this spurious cue when predicting answers.
- *Style Bias*: The text style is a kind of pattern that is independent of semantics. Models have learned to rely on the erroneous text style as a shortcut rather than capturing the underlying semantics. Adversaries can use this style bias to launch adversarial attacks [29].

2.2 Generalization and Robustness Challenge

The shortcut learning behavior could significantly hurt LLMs' **OOD generalization** as well as **adversarial robustness**. First, shortcut learning may result in significant performance degradation for OOD data. A common assumption is that training and test data are independently and identically distributed (IID). When LLMs are deployed in real-world applications with distribution shifts, this IID assumption will not hold any longer. These data typically do not contain the same types of bias and artifacts as the training data.

$$\begin{aligned} &\text{IID: } P_{train}(X, Y) = P_{test}(X, Y) \\ &\text{OOD: } P_{train}(X, Y) \neq P_{test}(X, Y) \end{aligned} \tag{1}$$

Using BERT-base as an example, there is a more than 20% reduction in accuracy on the OOD test set compared to the accuracy on the in-distribution test sets for NLU tasks [10]. To some extent, these models have solved the dataset rather than the underlying task. Second, shortcut learning produces models that are easily fooled by adversarial samples, which are generated when small and often imperceptible human-crafted perturbations are added to the normal input. One typical example is for the multiple choice reading comprehension task [39]. BERT models are attacked by adding distracting information, resulting in a significant performance drop. Further analysis indicates that these models are highly driven by superficial patterns, which inevitably leads to their adversarial vulnerability.

3 SHORTCUT LEARNING DETECTION

In this section, we discuss methods to identify shortcut learning problems in NLU models.

3.1 Comprehensive Performance Testing

Traditional evaluations employ IID training-test split of data. The test sets are drawn from the same distribution as the training sets and thus share the same kind of biases as the training data. Models

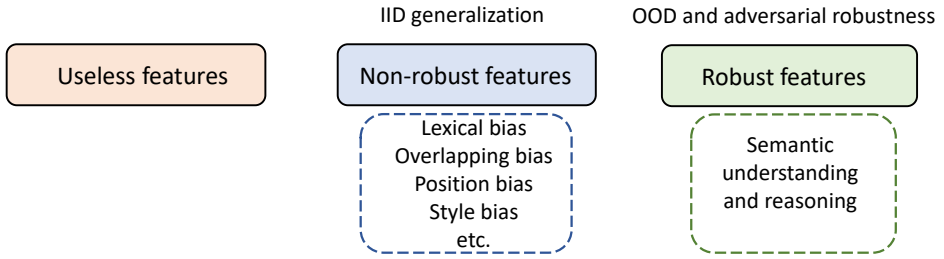

Figure 2: Features can be generally grouped into useless features, robust features, and non-robust features. Non-robust features indicate various kinds of biases captured by the model, which are not robust in the OOD setting. In contrast, robust features denote features of high-level semantic understanding that are robust to changes in the input.

that simply rely on memorizing superficial patterns could perform acceptablely on the IID test set. This type of evaluation has failed to identify the shortcut learning problem. Therefore, it is desirable to perform more comprehensive tests beyond the traditional IID testing.

First, the OOD generalization test has been proposed as an alternative to the IID test. Take the multi-genre natural language inference (MNLI) task for example, the HANS evaluation set is proposed to evaluate whether NLI models have syntactic heuristics: the lexical overlap heuristic, the subsequence heuristic, and the constituent heuristic [21]. Similarly, for the fact verification task, a symmetric test set is constructed that shares a philosophy similar to HANS [34]. These OOD tests have revealed dramatic performance degradation and exposed the shortcut learning problem of state-of-the-art LLMs.

Second, adversarial attacks could also be implemented to test the robustness of LLMs. For example, adversarial attacks have been used to reveal statistical bias in machine reading comprehension models [17]. Besides, the adversarial examples created through TextFooler [15] are used to test the generalization of common sense reasoning models [3]. The results indicate that the models have learned non-robust features and fail to generalize towards the main tasks associated with the datasets.

Third, randomization ablation methods are proposed to analyze whether LLMs have used these essential factors to achieve effective language understanding. For example, word order is a representative one among these significant factors. Recent ablation results indicate that word order does not matter for pre-trained language models [40]. In particular, LLMs are pre-trained first on sentences with randomly shuffled word order and then fine-tuned on various downstream tasks. The results show that these models still achieve high accuracy. Similarly, another study [27] has observed that LLMs are insensitive to word order in a wide set of tasks, including the entire GLUE benchmark. These experiments indicate that LLMs have ignored the syntax when performing downstream tasks, and their success can almost be explained by their ability to model higher-order word co-occurrence statistics.

3.2 Explainability Analysis

Model explainability is another effective tool that the community has used to identify the shortcut learning problem. LLMs are usually considered black boxes, as their decision-making process is opaque and difficult for humans to understand. This presents challenges in identifying whether these models make decisions based on justified reasons or on superficial patterns. Explainability enables us to diagnose spurious patterns captured by LLMs.

The existing literature mainly employs the explanation in the format of feature attribution to analyze shortcut learning behavior in NLU models [9]. Feature attribution is the most representative paradigm among all explainability-based methods. In particular, for each token x_i within a specific input x, the feature attribution algorithm ψ will calculate the contribution score ψ_i, which denotes the contribution score of that token for model prediction. For example, the Integrated Gradient [43] interpretation method is used to analyze the model behavior of BERT-based models [9]. It is observed that LLMs rely on dataset artifacts and biases within the hypothesis sentence for prediction, including functional words, negation words, etc [9]. This shortcut learning behavior is summarized further using the long-tailed phenomenon. Specifically, the tokens in the training set could be modeled using a long-tailed distribution. The LLM models concentrate mainly on information on the head of the distribution, which typically corresponds to non-generalizable shortcut tokens. In contrast, the tail of the distribution is poorly learned, although it contains abundant information for an NLU task.

Beyond feature attribution, other types of explainability methods have also been used to analyze shortcut learning behaviors. For example, instance attribution methods have been used to explain model prediction by identifying influential training data, which can be used to explain decision making logic for the current sample of interest [14]. Empirical analysis indicates that the most influential training data share similar artifacts, e.g., high overlap between the premise and hypothesis for the NLI task. Furthermore, hybrid methods that combine feature attribution and instance attribution have also been used to identify artifacts in the data [26]. The resulting explanations have provided a more comprehensive perspective on the shortcut learning behavior of LLMs.

4 ORIGINS OF SHORTCUT LEARNING

The problem of learning shortcuts in LLM models for NLU tasks is a result of multiple factors present in the training pipeline (see Figure 3). In this section, we will delve into these reasons and give particular emphasis to three key elements: the training datasets, the LLM model, and the fine-tuning training procedure.

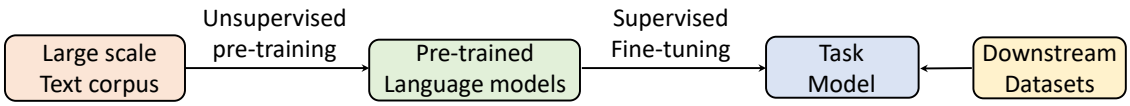

Figure 3: The pre-training then fine-tuning training paradigm. Shortcut learning can be attributed to different factors in this pipeline, including pre-trained language models, fine-tuning process, and downstream tasks.

4.1 Skewed Training Dataset

From a data standpoint, the NLU models' shortcut learning can be traced back to the annotation and collection artifacts of the training data to a large extent. Here, the training data include both the pre-training datasets as well as the downstream datasets (see Figure 3). Training sets are typically built through the crowd-sourcing process, which has the advantage of being low-cost and scalable. However, the crowd-sourcing process results in collection artifacts, where the training data are imbalanced with respect to features and class labels. Furthermore, when crowd workers author parts of the samples, they produce certain patterns of artifacts, i.e., annotation artifacts [13, 33]. Taking NLI as an example, the average sentence length of the hypothesis branch is shorter for the entailment category compared to the neutral category [13]. This suggests that crowd workers tend to remove words from the premise to create a hypothesis for the entailment category, leading to the overlap bias in the training data. Models trained on the skewed datasets will capture these artifacts and even amplify them during inference time.

4.2 LLMs Models

The robustness of NLU models is highly relevant to the pre-finetuned language models. In particular, there are two key factors: model sizes (measured by the number of parameters) and pre-training objectives.

First, models with the same kind of architectures and pre-training objective but with different sizes could have significantly different generalization ability. It is shown that increasing the size of the model could lead to an increase in the representation power and generalization ability. From the empirical perspective, comparisons have been made between LLMs of different sizes but with the same architecture, e.g., BERT-base with BERT-large, RoBERTa-base with RoBERTa-large [2, 46]. The results show that the larger versions generalize consistently better than the base versions, with a relatively smaller accuracy gap between the OOD and IID test data. Smaller models have fewer parameters than larger models and have a smaller model capacity. Therefore, smaller models are more prone to capture spurious patterns and are more dependent on data artifacts for prediction. Another work [10] studies the impact of model compression on the generalizability of LLMs and finds that compressed LLMs are significantly less robust compared to their uncompressed counterparts. Compressed models with knowledge distillation have also been shown to be more vulnerable to adversarial attacks. From a theoretical perspective, a recent analysis supports that there is a trade-off between the size of a model and its robustness, where large models tend to be more robust than smaller ones [5].

Second, LLMs with similar model sizes but with different pre-training objectives also differ in the generalization ability. Here, we consider three kinds of LLMs: BERT, ELECTRA, and RoBERTa. BERT is trained with masked language modeling and next-sentence prediction. RoBERTa removes the next-sentence prediction from BERT and uses dynamic masking. ELECTRA is trained to distinguish between real input tokens and fake input tokens generated by another network. Empirical analysis shows that these three models have significantly different levels of robustness [28]. For the Adversarial NLI (ANLI) dataset, it is shown that ELECTRA and RoBERTa have significantly better performance than BERT, for both the base and the large versions. Similarly, another study has shown that RoBERTa-base outperforms BERT-base around 20% in terms of accuracy on the HANS test set [2]. Because different architectures have distinct object functions during the pre-training stage, different inductive biases may be encoded by the models. This could possibly explain their differences in generalizability.

4.3 Model Fine-tuning Process

The learning dynamics could reveal what knowledge has been learned during the course of model training. There are some observations. First, standard training procedures have a bias toward learning simple features, which we can refer to as the simplicity bias. The models are based mainly on the simplest features and remain invariant to complex predictive features. Moreover, it has been observed that the models give overconfident predictions for easy samples and low-confidence predictions for hard samples. Second, models tend to learn non-robust and easy-to-learn features at the early stage of training. For example, reading comprehension models have learned the shortcut in the first few training iterations, which has influenced further exploration of the models for more robust features [17]. Third, it has been experimentally validated that longer fine-tuning could lead to better generalization. Specifically, a larger number of training epochs will dramatically improve the generalizability of LLMs in NLU tasks [46]. The preference for non-robust features can be explained from the following perspective. The present LLM training methods can be considered as data-driven, corpus-based, statistical, and machine learning approaches. It is postulated that while this data-driven paradigm may prove effective in certain NLP tasks, it falls short in relevance to the challenging NLU tasks that necessitate a deeper understanding of natural language.

5 MITIGATION OF SHORTCUT LEARNING

In this section, we introduce approaches that alleviate the problem of shortcut learning. The ultimate goal is to improve OOD generalization and adversarial robustness while still exhibiting good predictive performance in IID datasets. These methods are motivated mainly by the insights obtained in the last section. In particular, Section 5.1 introduces methods based on dataset refinement, and Section 5.2 focuses on model-centric mitigation approaches.

5.1 Data-Centric Mitigation Approaches

Dataset Refinement. Dataset refinement falls into the pre-processing mitigation family, with the aim of alleviating biases in the training datasets. First, when constructing new datasets, crowd workers will receive additional instructions to discourage the use of words that are highly indicative of annotation artifacts. Second, debiased datasets can also be developed by filtering out bias in existing data. For example, adversarial filtering is used to build a large-scale data set for the NLI task to reduce annotation artifacts that can be easily detected by a committee of strong baseline methods [54]. As a result, models trained on this dataset have to learn more generalizable features and rely on common sense reasoning to succeed. Third, we can also reorganize the train and test split, so that the bias distribution in the test set is different from that in the training set. Lastly, various types of data augmentation methods have been proposed. Representative examples include counterfactual data augmentation, mixup data augmentation, syntactically informative example augmentation by applying syntactic transformations to sentences, etc.

However, a drawback of this approach is that refining the dataset can only mitigate a limited number of recognized biases. The refined training set may not be completely free of biases and may still encompass statistical biases that are challenging for humans to identify. Thus, this could still negatively impact the model's performance.

Training Samples Reweighting. The main idea of reweighting is to place higher training weights on hard training samples, and vice versa [34, 47]. It is also called *worst-group loss minimization* in some literature. The underlying assumption is that improving the performance of the worst group (hard samples) is beneficial to the robustness of the model. It is typically achieved through two-stage training. In the first stage, the weight indexing model is trained; and in the second stage, the predictions of the indexing model are used as weights to adjust the importance of a training instance. Both soft weights [47] and hard weights could be used in the second stage. Another representative example is focal loss, which is based on a regularizer to assign higher weights to hard samples that have less confident predictions.

Partitioning Data into Environments. This line of methods follows the principle of invariant risk minimization [1], which encourages models to learn invariants in multiple environments. For example, training data has been partitioned into several non-IID subsets (i.e., training environments), where spurious correlations vary across environments and reliable ones remain stable across environments [45]. The training scheme is designed to encourage the model to rely on stable correlations and suppress spurious correlations. Another work proposes an inter-environment matching objective by maximizing the inner product between gradients from different environments, with the goal of increasing model generalization [37].

5.2 Model-Centric Mitigation Methods

In this section, we introduce model-centric mitigation methods, which can be named **robust learning** methods. These methods typically augment the traditional ERM-based training paradigm with different degrees of prior knowledge, explicitly or implicitly suppressing the model from capturing non-robust features. Some mitigation methods require that the shortcuts be known a priori, while others assume that the shortcuts are unknown.

Adversarial Training. This aims to learn better representations that do not contain information about artifacts or bias in the data. It is typically implemented in two ways in the NLP domain [31, 42]. First, the task classifier and adversarial classifier jointly share the same encoder [42]. The goal of the adversarial classifier is to provide the correct predictions for the artifacts in the training data. Then the encoder and task classifier can be trained to optimize the task objective while reducing the performance of the adversarial classifier in predicting artifacts. Second, adversarial examples are generated to maximize a loss function, and the model is trained to minimize the loss function. For example, the generator based on the masked language model is used to perturb the text to generate adversarial samples [31]. Despite the difference, both methods leverage the MinMax formulation during the debiasing process.

Explanation Regularization. This category aims to regularize model training using prior knowledge established by humans [19]. Specifically, it is achieved by regularizing the feature attribution explanations with rationale annotations created by domain experts, to enforce the model to make the right predictions for the right reasons [19]. These systems are trained to explicitly encourage the network to focus on features in the input that humans have annotated as important and suppress the models' attention to superficial patterns. For the NLI task, natural language explanations have been used to supervise the models, to encourage the model to pay more attention to the words present in the explanations [41]. It has significantly improved the models' OOD generalization performance. Note that this type of method can only be used when prior knowledge is known in advance about shortcuts.

Product-of-Expert (PoE). The goal is to train a debiased model by training it as an ensemble with a bias-only model [7]. This paradigm usually contains two stages. In the first stage, a bias-only model is explicitly trained to capture the bias of the data set, e.g. the hypothesis-only bias for the NLI task. During the second stage, the debiased model will be trained using cross-entropy loss, by combining its output with the output of the bias-only model: $\hat{p}_i = \text{softmax}\left(\log\left(p_i\right) + \log\left(b_i\right)\right)$. The parameters of the bias-only model are fixed during this stage, and only the debiased model parameters are updated by backpropagation. The goal is to encourage the debiased model to utilize orthogonal information with information from the bias-only model to make predictions.

Confidence Regularization. This mitigation scheme regularizes confidence in the model output, with the aim of encouraging the debiased model to give a higher uncertainty (lower confidence) for these biased samples. It is based on the observation that models tend to make overconfident predictions on biased examples. This relies on the training of a bias-only model to quantify the degree of bias of each training sample. The debiasing process is typically achieved through the knowledge distillation framework. In the first stage, the biased teacher model is trained using standard ERM loss, and the bias degree obtained from the bias-only model will be used to rescale the output distribution of the teacher model. In the second stage, the smoothed confidence values of the teacher model can be used to guide the training of the debiased model [9].

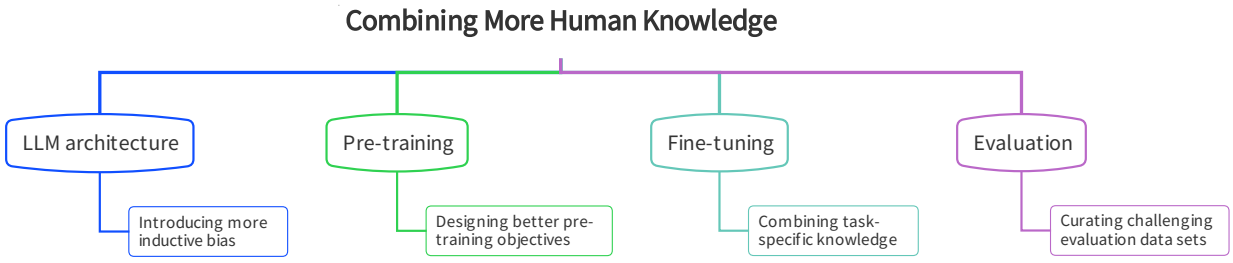

Figure 4: Combining more human knowledge to different stages of the pipeline. Specifically, knowledge can be combined to the architecture of model, model training process (both pre-training and fine-tuning), and model evaluation process.

Contrastive Learning. Contrastive learning can be used to guide the training of representations. The goal is to construct the instance discrimination task to guide the model to capture the robust and predictive features, while suppressing the undesirable non-robust features. The instance discrimination task should be carefully designed; otherwise, it is possible to suppress robust predictive features [32]. A representative work presents a framework for mitigating spurious correlations using contrastive learning [6]. The method synthesizes a pair of factual and counterfactual inputs from the original text by masking identified causal and non-causal terms respectively. The model learns to associate the causal term with task labels by comparing the original text with its counterfactual counterpart, while learning to ignore non-causal features by contrasting with the factual pair. The framework leads to models that are less dependent on non-robust features and exhibit improved generalization performance.

5.3 IID and Robustness Trade-off?

Another open question is about the connection between IID performance and OOD robustness performance. To the best of our knowledge, there are no consistent observations. For example, there is a linear correlation between IID performance and OOD generalization for different types of models introduced in Section 4.2. On the contrary, most robust learning methods introduced in Section 5.2 will sacrifice IID performance, although some of them could preserve IID performance. It deserves further research on the conditions under which the trade-off would occur. These insights could help the research community design robust learning frameworks that can simultaneously improve OOD and IID performance.

6 FUTURE RESEARCH DIRECTIONS

Despite the progress described in the previous sections, there are still numerous research challenges. In this section, we discuss potential research directions that could be pursued by the community.

6.1 Introducing More Domain Knowledge

Current standard of LLM training is data-driven. This is problematic because the resulting models essentially perform low-level pattern recognition. It may be useful for low-level NLP tasks like named-entity recognition (NER), but it is nearly impossible to tackle the more difficult natural language understanding tasks. As a result, it is preferable to combine the data-driven scheme with domain knowledge by incorporating knowledge at various stages of training. Furthermore, more knowledge should be applied to the design of the model architecture and the model evaluation (see Figure 4).

Inductive Bias to LLMs Models. It is suggested to introduce more inductive bias into the model architecture to improve robustness and generalization beyond IID benchmark datasets. Recently, some work has begun to induce certain kinds of linguistic structure in neural architectures. For example, TableFormer is proposed for robust table understanding [53]. It proposes a structurally aware table-text encoding architecture, where tabular structural biases are incorporated through learnable attention biases. Although introducing linguistic-oriented biases to the model architectures might not result in the best performance for benchmark datasets, it is essential to improve generalization beyond IID benchmarks. Note that inductive biases are highly task-dependent and should be carefully designed for each specific task to accommodate its unique characteristic.

Better Pre-training Objectives. The pre-training objective also plays a crucial role in determining the OOD robustness of fine-tuned language models. As an example, recent studies have shown that pre-trained BERT embeddings suffer from strong anisotropy, meaning the average cosine similarity is significantly higher than zero and word vectors cluster in narrow cones in the vector space [12, 18]. This leads to word representations having a high similarity to unrelated words, impacting their expressive power and accuracy in downstream tasks. It is desirable to invest more effort in designing better pre-training objectives to improve model robustness. Recent studies indicate that choosing a better pre-trained model could bring much better generalization performance than robust learning methods as introduced in Section 5. For example, RoBERTa-base with a standard fine-tuning loss could even outperform the BERT-base with robust learning objectives in terms of generalization performance on the HANS test set [2]. This highlights the importance of pre-training in NLU model generalization performance and calls for increased community efforts to improve pre-trained language models.

Better Fine-tuning Approaches. NLU tasks may contain various types of bias, which are not fully known even by domain experts. This is distinct from the literature that works with the toy task (e.g., Colored MNIST [1]), which typically contains a single type of bias and the bias is fully known. As a result, the majority of existing mitigation methods for NLU tasks rely on human prior knowledge heuristics. Some examples include: i) weak models are more prone to

capture biases, ii) non-robust models tend to give overconfident predictions for easy samples, etc. Unfortunately, this prior knowledge can only identify a limited number of biases in the data. Although it is possible to reduce the use of some identified shortcuts, models may still use other shortcuts for prediction. This could explain why existing mitigation methods only provide a limited improvement in generalization. As a result, it is suggested to incorporate more human-like common sense knowledge into the model training.

Curating Challenging Evaluation Datasets. It is encouraging to see that some benchmark datasets for adversarial and OOD robustness have emerged. For example, adversarial GLUE is proposed for adversarial robustness evaluation, which contains 14 adversarial attack methods [50]. Despite these recent advances, it is necessary to continue curating difficult evaluation datasets that cover a wider range of NLU tasks, such as reading comprehension, and that cover a wider range of biases, such as those listed in Section 2.1.

6.2 Revisiting The Mitigation Approaches

Existing mitigation methods have typically had limited mitigation performance. For example, for the MNLI task, the accuracy for mitigated models with BERT-base as the backbone is consistently lower than 70% for the HANS test set [47]. Note that HANS is a balanced binary test set, where 50% is the accuracy of the random guess. The improvement in performance falls far short of our expectations. This brings up the following questions: 1) What have the mitigation algorithms accomplished, and 2) how can mitigation performance be improved further?

Debiased algorithms are thought to achieve better generalization because they can learn more robust features than biased models that rely primarily on non-robust features. However, this is not always the case with debiased algorithms. A recent work uses explainability as a debugging tool to analyze debiased models [22]. The analysis indicates that the debiased models actually encode more biases in their inner representations. It is speculated that the improved performance on the OOD data comes from the refined classification head. More research is needed to investigate whether the debiased model has captured more robust features and what is the source of their improved generalization. This also suggests an interesting research direction by only updating the biased classification head, as updating the entire model is typically difficult and time consuming.

6.3 In-depth Theoretical Understanding

In addition to the current empirical research, there is also a growing trend of preliminary theoretical research aimed at uncovering the shortcut learning behavior of DNN models [23, 36, 49]. For instance, using one-hidden-layer neural networks as the base model, one theoretical work uncovers that neural networks tend to exclusively rely on simplest and non-robust features, while remain invariant to other useful but more complex features [36]. This type of *simplicity bias* is one of the primary causes of low OOD generalization and adversarial vulnerability. Another theoretical study has investigated the reason behind superficial correlations from the optimization perspective [49]. By using a depth-2 ReLU network as an example, the study proposed the *Gradient Starvation* phenomenon, which states that the gradient descent optimization methods tend to learn non-robust networks while slowing down the learning of robust and task-relevant features. Although these existing works provide insights into the reason of shortcut learning of shallow neural networks, there is still a lack of a solid theoretical understanding of why LLMs learn shortcuts. In the future, further research is needed to fully explain this tendency in the context of LLMs.

6.4 Taking Inspiration from Other Directions

In addition, we can take inspiration from other relevant directions to address the shortcut learning issue of LLMs.

Domain Adaptation & Generalization. The robust learning approaches that we have discussed in Section 5 are closely relevant to domain adaptation and domain generalization. The three directions share the similarity that the training and test sets are not from the same distribution, i.e., there is a certain distribution shift. However, the objective of robust learning is distinct from domain adaptation, which aims to generalize to a specific target domain. In contrast, robust learning is closer to domain generalization, where both areas have the goal of generalizing over a range of unknown conditions. The NLP community can leverage the findings from the domain generalization area to design more robust learning methods for LLMs.

Long-Tailed Classification. Long-tailed classification addresses the issue of long-tailed distributed data, in which the head class has a large number of training samples while the tail class has few. Shortcut learning can be treated as a special case of long-tailed classification, where easy samples correspond to the head class and hard samples represent the tail class. Some of the robust learning solutions (e.g., reweighting) in Section 5 share a similar philosophy with approaches to the long-tailed classification problem. Leveraging ideas from approaches to long-tailed classification could improve the robustness of LLMs even further.

Algorithmic Discrimination. Shortcut learning could also lead to discrimination and unfairness in deep learning models. In contrast to the general bias captured by the models, the spurious patterns here usually correspond to societal biases in terms of humans (e.g., racial bias and gender bias) [11]. Here, the models have associated the fairness-sensitive attributes (e.g., ZIP code and surname) with main prediction task labels (e.g., mortgage loan rejection). At the inference time, the model would amplify the bias and show discrimination towards certain demographic groups, e.g., African Americans.

6.5 Motivating Other Directions

We can also take advantage of the insights discussed above to motivate the development of other directions.

Backdoor Attack. The previous sections focus on discussing the setting in which LLMs have unintentionally captured undesirable shortcuts. However, the adversary can intentionally insert shortcuts into LLMs, which could be a potential security threat to the deployed LLMs. This is termed the backdoor attack (or poisoning/Trojan attack) [44]. Backdoor attackers insert human-crafted easy patterns that serve as shortcuts during the model training process, explicitly encouraging the model to learn shortcuts. Representative examples include modifying the style of text, adding shortcut unigrams such as double quotation marks, etc.

Watermarking. Unlike malicious use of shortcut learning as the backdoor attack, shortcut learning can also be used for benign purposes. In particular, trigger patterns can be inserted as watermarks by model owners during the training phase to protect the IP of companies. When LLMs are used by unauthorized users, shortcuts in the format of trigger patterns can be used by the stakeholders to claim ownership of the models.

7 PROMPT-BASED PARADIGM

In previous sections, we have explored the characterization of the shortcut learning problem in the pre-training and fine-tuning training paradigm of medium-sized language models (typically with less than a billion parameters). With the recent emergence of huge-sized language models (with billions of parameters) such as GPT-3 and T5, the prompt-based paradigm has evolved into a new training paradigm with distinct formats from the standard fine-tuning paradigm. Consider the example of prompt for GPT-3. Using natural language instructions and/or demonstration of a few tasks, the LLM can generate the desired output without the need for gradient updates or fine-tuning. In this section, we examine the robustness of prompt-based methods and then compare them with the traditional standard fine-tuning approach.

7.1 Robustness of Prompt-based Methods

There are two types of prompt-based paradigms: 1) prompt-based fine-tuning and 2) prompting without fine-tuning. Prompt-based fine-tuning aims to enable medium-sized language models like BERT or RoBERTa to be few-shot learners, and this still requires optimizing the model's parameters. On the other hand, prompting without fine-tuning is meant for huge-sized language models like GPT-3, where the parameters are fixed and the model is applied to various tasks using different prompts, either discrete or soft. In the following discussion, we will discuss the shortcut learning issue in both types of prompt-based paradigms.

Prompt-based Fine-tuning. Preliminary research has been conducted to examine the shortcut learning challenge in the few-shot prompt-based fine-tuning paradigm [48]. This preliminary study investigated the RoBERTa-large model, which comprises 355 million parameters. This work reveals the following insights: (i) zero-shot prompt-based models exhibit a higher level of robustness against the lexical overlap heuristic during inference, as evidenced by their strong performance on relevant challenge datasets. (ii) conversely, prompt-based finetuned models tend to adopt the spurious heuristic as they learn from larger amounts of labeled data, which is reflected by poor performance on OOD datasets. This indicates that prompt-based fine-tuning negatively impacts the robustness and generalizability of a model, just like the standard fine-tuning. The primary reason is that both training methods require adjusting the model's parameters using a biased NLI dataset, leading to a model that heavily relies on dataset biases as shortcuts for predictions.

Prompting Without Fine-tuning. Preliminary studies are emerging to examine the robustness of prompt-based methods for huge-size language models [51, 55]. A study examines the few-shot learning performance of GPT-3 (2.7B, 13B, and 175B parameters) and GPT-2 (1.5B parameters) on text classification and information extraction tasks [55]. The results of the analysis reveal that the investigated LLMs are susceptible to majority label bias and position bias, where they tend to predict answers based on the frequency or position of the answers in the training data. Additionally, these LLMs also exhibit common token bias, where they favor answers that are prevalent in their pre-training corpus. Another study explores the impact of prompts on natural language inference tasks in zero-shot and few-shot settings using T0 (3B and 11B parameters) and GPT-3 (175B parameters) [51]. Experimental results suggest that models can learn just as quickly with many irrelevant or even misleading prompts as they can with effective and instructive prompts. This indicate that models' improvement is not derived from models understanding task instructions in ways analogous to humans' use of task instructions.

7.2 Prompting versus Standard Fine-tuning

GPT-3's few-shot prompt performance is compared to that of BERT and RoBERTa through standard fine-tuning on two natural language inference tasks, i.e., MNLI and QQP [38]. Additionally, these models are evaluated on the corresponding difficult OOD datasets: HANS and PAWS. The results show that GPT-3 performs slightly worse in generalization than BERT and RoBERTa on the in distribution MNLI and QQP datasets. On the other hand, GPT-3 achieves higher accuracy on the OOD tests for the majority of testing settings, indicating that GPT-3 has a lower generalization gap between the in distribution test set and the OOD test set, and thus a higher robustness. However, further analysis of the HANS dataset reveals that GPT-3 still exhibits substantial performance disparities between the bias-supporting and bias-countering subsets. This implies that there is room for enhancing the robustness of prompt-based techniques.

Note that the current research on prompt-based methods primarily aims at improving LLMs' performance on standard benchmarks. The robustness and generalization of this paradigm still require further investigation. A more thorough evaluation of prompt-based methods is needed and could be a future research topic. Additionally, techniques such as Chain-of-Thought [52] and Scratchpad [25] have been utilized to encourage models to perform intermediate calculations. These methods have proven to enhance the reasoning abilities of LLMs, thus having the potential to improve their robustness and generalization capabilities. Lastly, developing mitigation frameworks that can improve generalization performance on OOD test sets without sacrificing standard benchmark performance deserves more attention from the research community.

8 CONCLUSIONS

We present a thorough survey of the LLM's shortcut learning issue for NLU tasks in this article. Our findings suggest that shortcut learning is caused by a skewed dataset, model architecture, and model learning dynamics. We also summarize the mitigation solutions that can be used to reduce shortcut learning and improve the robustness of LLMs. Furthermore, we discuss directions that merit additional research effort from the research community, as well as the connections between shortcut learning and other relevant directions. The key takeaways from this survey's analysis are that the current pure data-driven training paradigm for LLMs is insufficient for high-level natural language understanding. In the future, the data-driven paradigm should be combined with domain knowledge at every stage of model design and evaluation to advance the field of LLMs.

REFERENCES

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

[2] Prajjwal Bhargava, Aleksandr Drozd, and Anna Rogers. 2021. Generalization in NLI: Ways (Not) To Go Beyond Simple Heuristics. In *Proceedings of the Second Workshop on Insights from Negative Results in NLP*.

[3] Ruben Branco, António Branco, João Silva, and João Rodrigues. 2021. Shortcutted Commonsense: Data Spuriousness in Deep Learning of Commonsense Reasoning. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

[4] Tom B Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in Neural Information Processing Systems (NeurIPS)* (2020).

[5] Sébastien Bubeck and Mark Sellke. 2021. A universal law of robustness via isoperimetry. *Advances in Neural Information Processing Systems (NeurIPS)* (2021).

[6] Seungtaek Choi, Myeongho Jeong, Hojae Han, and Seung-won Hwang. 2022. C2l: Causally contrastive learning for robust text classification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 36. 10526–10534.

[7] Christopher Clark, Mark Yatskar, and Luke Zettlemoyer. 2019. Don't Take the Easy Way Out: Ensemble Based Methods for Avoiding Known Dataset Biases. *Empirical Methods in Natural Language Processing (EMNLP)* (2019).

[8] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. *North American Chapter of the Association for Computational Linguistics (NAACL)* (2019).

[9] Mengnan Du, Varun Manjunatha, Rajiv Jain, Ruchi Deshpande, Franck Dernoncourt, Jiuxiang Gu, Tong Sun, and Xia Hu. 2021. Towards Interpreting and Mitigating Shortcut Learning Behavior of NLU Models. *North American Chapter of the Association for Computational Linguistics (NAACL)* (2021).

[10] Mengnan Du, Subhabrata Mukherjee, Yu Cheng, Milad Shokouhi, Xia Hu, and Ahmed Hassan Awadallah. 2023. Robustness Challenges in Model Distillation and Pruning for Natural Language Understanding. *The 17th Annual Meeting of the European chapter of the Association for Computational Linguistics (EACL)* (2023).

[11] Mengnan Du, Fan Yang, Na Zou, and Xia Hu. 2020. Fairness in deep learning: A computational perspective. *IEEE Intelligent Systems* (2020).

[12] Kawin Ethayarajh. 2019. How Contextual are Contextualized Word Representations? Comparing the Geometry of BERT, ELMo, and GPT-2 Embeddings. 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)*. 55–65.

[13] Suchin Gururangan, Swabha Swayamdipta, Omer Levy, Roy Schwartz, Samuel R Bowman, and Noah A Smith. 2018. Annotation artifacts in natural language inference data. *North American Chapter of the Association for Computational Linguistics (NAACL)* (2018).

[14] Xiaochuang Han, Byron C Wallace, and Yulia Tsvetkov. 2020. Explaining Black Box Predictions and Unveiling Data Artifacts through Influence Functions. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL)*.

[15] Di Jin, Zhijing Jin, Joey Tianyi Zhou, and Peter Szolovits. 2020. Is bert really robust? a strong baseline for natural language attack on text classification and entailment. In *Proceedings of the AAAI conference on artificial intelligence (AAAI)*.

[16] Miyoung Ko, Jinhyuk Lee, Hyunjae Kim, Gangwoo Kim, and Jaewoo Kang. 2020. Look at the First Sentence: Position Bias in Question Answering. In *Empirical Methods in Natural Language Processing (EMNLP)*.

[17] Yuxuan Lai, Chen Zhang, Yansong Feng, Quzhe Huang, and Dongyan Zhao. 2021. Why Machine Reading Comprehension Models Learn Shortcuts? *ACL Findings* (2021).

[18] Yuxin Liang, Rui Cao, Jie Zheng, Jie Ren, and Ling Gao. 2021. Learning to remove: Towards isotropic pre-trained BERT embedding. *Artificial Neural Networks and Machine Learning–ICANN 2021: 30th International Conference on Artificial Neural Networks, Bratislava, Slovakia, September 14–17, 2021, Proceedings, Part V 30* (2021), 448–459.

[19] Frederick Liu and Besim Avci. 2019. Incorporating priors with feature attribution on text classification. *57th Annual Meeting of the Association for Computational Linguistics (ACL)* (2019).

[20] 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. *arXiv preprint arXiv:1907.11692* (2019).

[21] R Thomas McCoy, Ellie Pavlick, and Tal Linzen. 2019. Right for the wrong reasons: Diagnosing syntactic heuristics in natural language inference. *57th Annual Meeting of the Association for Computational Linguistics (ACL)* (2019).

[22] Michael Mendelson and Yonatan Belinkov. 2021. Debiasing Methods in Natural Language Understanding Make Bias More Accessible. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

[23] Depen Morwani, Jatin Batra, Prateek Jain, and Praneeth Netrapalli. 2023. Simplicity Bias in 1-Hidden Layer Neural Networks. *arXiv preprint arXiv:2302.00457* (2023).

[24] Timothy Niven and Hung-Yu Kao. 2019. Probing neural network comprehension of natural language arguments. *57th Annual Meeting of the Association for Computational Linguistics (ACL)* (2019).

[25] Maxwell Nye, Anders Johan Andreassen, Guy Gur-Ari, Henryk Michalewski, Jacob Austin, David Bieber, David Dohan, Aitor Lewkowycz, Maarten Bosma, David Luan, et al. 2022. Show Your Work: Scratchpads for Intermediate Computation with Language Models. *Deep Learning for Code Workshop* (2022).

[26] Pouya Pezeshkpour, Sarthak Jain, Sameer Singh, and Byron C Wallace. 2021. Combining feature and instance attribution to detect artifacts. *arXiv preprint arXiv:2107.00323* (2021).

[27] Thang M Pham, Trung Bui, Long Mai, and Anh Nguyen. 2020. Out of Order: How important is the sequential order of words in a sentence in Natural Language Understanding tasks? *arXiv preprint arXiv:2012.15180* (2020).

[28] Grusha Prasad, Yixin Nie, Mohit Bansal, Robin Jia, Douwe Kiela, and Adina Williams. 2021. To what extent do human explanations of model behavior align with actual model behavior?. In *Proceedings of the Fourth BlackboxNLP Workshop on Analyzing and Interpreting Neural Networks for NLP*.

[29] Fanchao Qi, Yangyi Chen, Xurui Zhang, Mukai Li, Zhiyuan Liu, and Maosong Sun. 2021. Mind the Style of Text! Adversarial and Backdoor Attacks Based on Text Style Transfer. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

[30] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2020. Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer. *Journal of Machine Learning Research (JMLR)* (2020).

[31] Ahmad Rashid, Vasileios Lioutas, and Mehdi Rezagholizadeh. 2021. MATE-KD: Masked Adversarial TExt, a Companion to Knowledge Distillation. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (ACL)*.

[32] Joshua Robinson, Li Sun, Ke Yu, Kayhan Batmanghelich, Stefanie Jegelka, and Suvrit Sra. 2021. Can contrastive learning avoid shortcut solutions? *Advances in Neural Information Processing Systems (NeurIPS)* (2021).

[33] Michael Saxon, Xinyi Wang, Wenda Xu, and William Yang Wang. 2023. PECO: Examining Single Sentence Label Leakage in Natural Language Inference Datasets through Progressive Evaluation of Cluster Outliers. In *Proceedings of the 17th Conference of the European Chapter of the Association for Computational Linguistics*. 3053–3066.

[34] Tal Schuster, Darsh J Shah, Yun Jie Serene Yeo, Daniel Filizzola, Enrico Santus, and Regina Barzilay. 2019. Towards debiasing fact verification models. *Empirical Methods in Natural Language Processing (EMNLP)* (2019).

[35] Priyanka Sen and Amir Saffari. 2020. What do Models Learn from Question Answering Datasets?. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

[36] Harshay Shah, Kaustav Tamuly, Aditi Raghunathan, Prateek Jain, and Praneeth Netrapalli. 2020. The pitfalls of simplicity bias in neural networks. *Advances in Neural Information Processing Systems (NeurIPS)* (2020).

[37] Yuge Shi, Jeffrey Seely, Philip HS Torr, N Siddharth, Awni Hannun, Nicolas Usunier, and Gabriel Synnaeve. 2022. Gradient matching for domain generalization. *International Conference on Learning Representations (ICLR)* (2022).

[38] Chenglei Si, Zhe Gan, Zhengyuan Yang, Shuohang Wang, Jianfeng Wang, Jordan Boyd-Graber, and Lijuan Wang. 2022. Prompting gpt-3 to be reliable. *arXiv preprint arXiv:2210.09150* (2022).

[39] Chenglei Si, Shuohang Wang, Min-Yen Kan, and Jing Jiang. 2019. What does BERT Learn from Multiple-Choice Reading Comprehension Datasets? *arXiv preprint arXiv:1910.12391* (2019).

[40] Koustuv Sinha, Robin Jia, Dieuwke Hupkes, Joelle Pineau, Adina Williams, and Douwe Kiela. 2021. Masked language modeling and the distributional hypothesis: Order word matters pre-training for little. *Empirical Methods in Natural Language Processing (EMNLP)* (2021).

[41] Joe Stacey, Yonatan Belinkov, and Marek Rei. 2022. Supervising Model Attention with Human Explanations for Robust Natural Language Inference. *AAAI Conference on Artificial Intelligence (AAAI)* (2022).

[42] Joe Stacey, Pasquale Minervini, Haim Dubossarsky, Sebastian Riedel, and Tim Rocktäschel. 2020. Avoiding the Hypothesis-Only Bias in Natural Language Inference via Ensemble Adversarial Training. *Empirical Methods in Natural Language Processing (EMNLP)* (2020).

[43] Mukund Sundararajan, Ankur Taly, and Qiqi Yan. 2017. Axiomatic Attribution for Deep Networks. *International Conference on Machine Learning (ICML)* (2017).

[44] Ruixiang Tang, Mengnan Du, Ninghao Liu, Fan Yang, and Xia Hu. 2020. An embarrassingly simple approach for trojan attack in deep neural networks. In *Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining (KDD)*.

[45] Damien Teney, Ehsan Abbasnejad, and Anton van den Hengel. 2020. Unshuffling data for improved generalization. *arXiv preprint arXiv:2002.11894* (2020).

[46] Lifu Tu, Garima Lalwani, Spandana Gella, and He He. 2020. An empirical study on robustness to spurious correlations using pre-trained language models. *Transactions of the Association for Computational Linguistics (TACL)* (2020).

[47] Prasetya Ajie Utama, Nafise Sadat Moosavi, and Iryna Gurevych. 2020. Towards debiasing NLU models from unknown biases. *Empirical Methods in Natural Language Processing (EMNLP)* (2020).

[48] Prasetya Ajie Utama, Nafise Sadat Moosavi, Victor Sanh, and Iryna Gurevych. 2021. Avoiding Inference Heuristics in Few-shot Prompt-based Finetuning. *Empirical Methods in Natural Language Processing (EMNLP)* (2021).

[49] Gal Vardi, Gilad Yehudai, and Ohad Shamir. 2022. Gradient Methods Provably Converge to Non-Robust Networks. *arXiv preprint arXiv:2202.04347* (2022).

[50] Boxin Wang, Chejian Xu, Shuohang Wang, Zhe Gan, Yu Cheng, Jianfeng Gao, Ahmed Hassan Awadallah, and Bo Li. 2021. Adversarial GLUE: A Multi-Task Benchmark for Robustness Evaluation of Language Models. *Thirty-fifth Conference on Neural Information Processing Systems Datasets and Benchmarks Track (Round 2)* (2021).

[51] Albert Webson and Ellie Pavlick. 2022. Do Prompt-Based Models Really Understand the Meaning of Their Prompts?. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*.

[52] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed H Chi, Quoc V Le, Denny Zhou, et al. 2022. Chain-of-Thought Prompting Elicits Reasoning in Large Language Models. In *Advances in Neural Information Processing Systems*.

[53] Jingfeng Yang, Aditya Gupta, Shyam Upadhyay, Luheng He, Rahul Goel, and Shachi Paul. 2022. TableFormer: Robust Transformer Modeling for Table-Text Encoding. *60th Annual Meeting of the Association for Computational Linguistics (ACL)* (2022).

[54] Rowan Zellers, Yonatan Bisk, Roy Schwartz, and Yejin Choi. 2018. Swag: A large-scale adversarial dataset for grounded commonsense inference. *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing (EMNLP)* (2018).

[55] Zihao Zhao, Eric Wallace, Shi Feng, Dan Klein, and Sameer Singh. 2021. Calibrate before use: Improving few-shot performance of language models. In *International Conference on Machine Learning*. PMLR, 12697–12706.