Refining Models' Inference Ability Independent of Simple Bag-of-Word Statistics in Natural Language Inference

Anonymous EMNLP-IJCNLP submission

Abstract

Natural language inference (NLI) considers the task of determining the semantic inferential relationship (e.g., contradiction) between two sentences. While neural network-based NLI systems have achieved impressive results, recent literature suggests that these empirical successes are achieved through models' abilities in taking advantage of the distributional bias of some keywords in the corpus (e.g., a model might predict contradiction given the existence of word "no" in one input sentence). As a result, these models can predict with high accuracy without understanding the inference relationship as how the human defines it, while perform relatively poorly on out-of-domain samples that do not share the same superficial bag-of-words statistics with training samples. In this paper, we apply a technique that can mitigate the above issue by encouraging the model to discard the predictive signals learned directly through the bag-of-words distributions, which leads to equivalent or better performance on out-of-domain testing samples. To facilitate development of robust models less susceptible to such bag-of-words statistics, we also propose a systematic procedure to rearrange an existing NLI dataset, such that the testing samples cannot be easily answered by only exploiting bag-of-words features.

1 Introduction

Natural language inference (NLI) serves as one of the central research topics in the study of natural language understanding techniques (Dagan et al., 2005; MacCartney and Manning, 2009). As a machine learning task, NLI is a three-way classification of the inference relationships (entailment, neutral, contradiction) between two text fragments: a *premise* (e.g., *People are playing soccer*) and a *hypothesis* (e.g., *There is a sport game*) (Bowman et al., 2015). Recently, with the introduction of several large NLI corpus (Bowman et al., 2015; Williams et al., 2018), machine learning methods, particularly neural networks, have made tremendous progress and achieved impressive empirical performances. The high prediction accuracy seemingly indicates that the community has developed models that can perform the NLI task roughly at the human level.

However, a closer investigation suggests that existing NLI systems might manage to achieve high performance *without* actually understanding the underlying semantics of input sentences: Poliak et al. (2018) demonstrated that a model could get reasonably high prediction accuracy even if the model only sees the hypotheses. Also, when Wang et al. (2019b) swapped the premises and hypotheses, they observed (sometimes significant) performance drops across all inference relationships for many NLI models, while the labels are believed to preserve for some inference relationships. Similar performance drops are observed when Talman and Chatzikyriakidis (2018) tested the NLI models across different testing datasets. Additional empirical evidence has also been reported to show that NLI models may not learn how the human defines the NLI despite their high prediction accuracy (Glockner et al., 2018; Talman and Chatzikyriakidis, 2018; McCoy et al., 2019b).

Recently, Gururangan et al. (2018) suggest that the models' high performance (without truly understanding the semantic relationship between input sentences) is possibly due to that the model could exploit the distributional bias of some keywords in the corpus to infer the correct answer. As we will elaborate in § 2, such distributional bias of keywords is usually introduced through the crowdsourcing data collection process. For instance, a crowd worker could easily generate a *contradiction* hypothesis by simply adding the word *no* to the premise, creating a spurious association between the label (e.g., contradiction) and the superficial bag-of-words statistics in the hypothesis. A data-driven model could therefore take advantage of such superficial statistics to infer the correct answer without awareness of the actual semantic relationship of input sentences, while performing poorly on "adversarial" testing samples that do not share the same bag-of-words statistics with the training data.

In this paper we adapt the technique proposed by Wang et al. (2019a), and alleviate the above issue by forcing the model to predict with representations that are not explainable by these undesired signals (§ 3). The central assumption of this technique is that by forcing the model to depend less on the undesired signals, we can regularize the model to learn more about how the human defines the NLI relationship. Empirically, this approach improves the performance when tested on out-of-domain samples that do not share the same distributional bias as the training samples (§ 5).

Additionally, to facilitate future development of robust NLI systems less susceptible to superficial bag-of-words statistics, we also introduce a simple technique to generate benchmark datasets targeted to evaluate a model's robustness against such superficial signals. Specifically, we rearrange the examples in an existing NLI dataset, such that the newly-arranged test set is composed of samples that are least predictive from word-level statistics (§ 4). Comparing to existing NLI robustness evaluation methods (e.g., Naik et al., 2018; Gururangan et al., 2018; Glockner et al., 2018), which are usually manually created to test limited and specific aspects of a model, our method can automatically rearrange a given dataset to evaluate a model's inference ability beyond simple bag-of-words statistics. We put forward ReSNLI, a rearranged SNLI dataset (Bowman et al., 2015), and conduct experiments with several NLI models. Our results and error analysis suggest ReSNLI as a better territory to evaluate the textual inference relationship beyond bag-of-words statistical regularities (§ 5).

2 Background

Problem Definition NLI aims to determine the semantic relationship between a *premise* (P) sentence and a *hypothesis* (H) sentence. The three pre-defined relationships are:

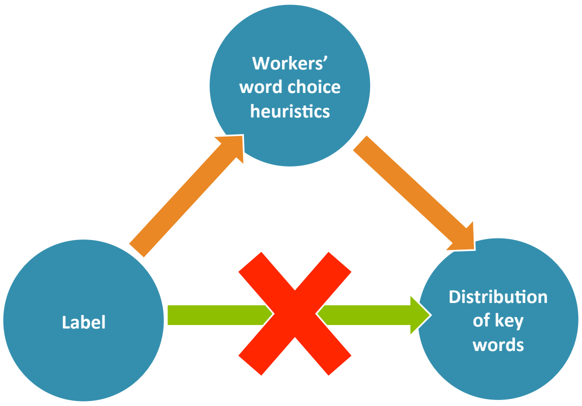

Figure 1: Illustration of how the Amazon Mechanical Turk workers' word choosing heuristics serve as the confounding factor that creates the spurious association between the NLI label and the distribution of words in *hypothesis*.

E (*Entailment*): H is true given P.

C (*Contradiction*): H is not true given P.

N (*Neutral*): H may or may not be true given P.

Confounding Signals One of the underlying reasons for a model's high in-domain performance and poor generalization ability is discussed in Gururangan et al. (2018) as the annotation artifacts of NLI data: the NLI data is generated in a way that Amazon Mechanical Turk workers are asked to write down the *hypothesis* when they are shown the *premise* and the label. Along with this process, these workers invented some heuristics that allow them to quickly finish the writing, such as directly injecting the word "no" or "never" into the *premise* to create a *hypothesis* with the label *contradiction*. These heuristics serve as the confounding factor and create some spurious association between the distribution of some words in the *hypothesis* and the label, as illustrated in Fig. 1. A model can be trained to exploit these confounding signals and result in high predictive accuracy when tested with samples from the same dataset, but lower performance when tested with out-of-domain samples.

Motivated by the fact that NLI models are exploiting these confounding signals, we apply a technique that can help the model depend less on these confounding signals, but more on understanding the sentences (§ 3). To facilitate development of robust models less susceptible to such bag-of-words statistics, we also propose a systematic procedure to rearrange an existing NLI dataset to make the testing samples harder to be answered by only exploiting bag-of-words statistics (§ 4).

3 Discard Confounding Signals

Set-up We use $X = \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_l\}$ to denote a (concatenated) sentence pair with l words, where $\mathbf{x}_i$ is a pre-trained embedding of x_i. Let y denote the corresponding NLI label. We use $\langle \mathcal{X}, \mathcal{Y} \rangle$ to denote the entire dataset, and $f(h(\mathbf{X};\theta);\delta)$ a generic NLI neural model, where $h(\cdot;\theta)$ is the encoder and $f(\cdot;\delta)$ the decoder. Note that "encoder" and "decode" are only conceptual, and for a representation extracted by any layer, we can call the neural architecture below this layer the "encoder" and regard those subsequent layers as the "decoder", which map this representation to the output logits. Thus, a generic optimization procedure for NLI can be written as follows:

$$\widehat{\theta}, \widehat{\delta} = \arg\min_{\theta,\delta} l(\widehat{\mathbf{y}}, \mathbf{y}) + R_\lambda,$$

where $\widehat{\mathbf{y}} = \arg\max f(h(\mathbf{X};\theta);\delta)$. l is a generic loss function (e.g., the commonly used cross-entropy loss), and R_λ is the regularization term. We follow the set-up of Wang et al. (2019a) to define the final classifier as "decoder" $f(\cdot;\delta)$ and any preceding neural architecture as "encoder" $h(\cdot;\theta)$.

Modeling Confounding Signals As discussed in § 2, the confounding signals introduced during the construction phase of the dataset are usually straightforward associations between the appearances of certain words and the label. We aim to use a simple statistical model to capture such confounding signals. Our underlying assumption is that, for a simple model without enough capacity to capture the semantic relationship between input sentences, it has to exploit such superficial signals in order to achieve high performance.

Therefore, to model the confounding signals, we build a simple one-layer neural network by taking the average of word embeddings from both premise and hypothesis as input:

$$g(X;\phi) = \mathrm{ReLU}(\mathbf{W}^\phi \frac{\sum_{i=1}^{l} \mathbf{x}_i}{l}).$$

Discarding Confounding Signals To force the model to depend less on the confounding signals learned through $g(\mathbf{X};\phi)$, we primarily follow the technique introduced by Wang et al. (2019a). Specifically, $f(\cdot;\delta)$ is extended to consume the concatenation of the NLI-model encoded representation and the confounding signals: $f([h(\mathbf{X};\theta), g(\mathbf{X};\phi)];\delta)$. We define:

$$\mathbf{F}_B = f([h(\mathbf{X};\theta), g(\mathbf{X};\phi)];\delta),$$
$$\mathbf{F}_G = f([\mathbf{0}, g(\mathbf{X};\phi)];\delta),$$

where $\mathbf{0}$ is a matrix of all zeros. We want to optimize the model through a representation (i.e., $\mathbf{F}_L$) as a function of $\mathbf{F}_B$ and $\mathbf{F}_G$, such that $\mathbf{F}_L$ cannot be explained by $\mathbf{F}_G$. Therefore, we can solve a linear system by minimizing how much $\mathbf{F}_G$ can explain $\mathbf{F}_B$: for k-th column in $\mathbf{F}_B$ (i.e., $\mathbf{F}_B^{(k)}$), we solve the problem:

$$\widehat{\omega_k} = \arg\min ||\mathbf{F}_B^{(k)} - \mathbf{F}_G \omega_k||_2^2, \qquad (1)$$

where ω_k stands for the regression coefficients. Then $\mathbf{F}_L^{(k)}$ can be achieved as the residue of $\mathbf{F}_B^{(k)}$ after $\mathbf{F}_B^{(k)}$ is explained by $\mathbf{F}_G$:

$$\mathbf{F}_L^{(k)} = \mathbf{F}_B^{(k)} - \mathbf{F}_G \widehat{\omega_k}.$$

There is a closed-form solution of Eq. 1:

$$\widehat{\omega_k} = (\mathbf{F}_G^T \mathbf{F}_G)^{-1} \mathbf{F}_G^T \mathbf{F}_B^{(k)},$$

which gives us the residual representation $\mathbf{F}_L$ not explainable from $\mathbf{F}_G$:

$$\mathbf{F}_L^{(k)} = \mathbf{F}_B^{(k)} - \mathbf{F}_G (\mathbf{F}_G^T \mathbf{F}_G)^{-1} \mathbf{F}_G^T \mathbf{F}_B^{(k)},$$
$$\mathbf{F}_L = (\mathbf{I} - \mathbf{F}_G (\mathbf{F}_G^T \mathbf{F}_G)^{-1} \mathbf{F}_G^T) \mathbf{F}_B.$$

Then the prediction is computed by:

$$\widehat{\mathbf{y}} = \arg\max \mathbf{F}_L.$$

For a fair comparison, we adopt the same regularization term R_λ as the original model. Following the convention in Wang et al. (2019a), this method is denoted as HEX.

4 Rearrange NLI Datasets for Robustness Evaluation

To evaluate a NLI model's robustness against superficial bag-of-words statistics, we propose a systematic approach to rearrange the samples in an existing NLI dataset, such that the rearranged testing set contains samples that cannot be easily answered using superficial word-level statistics. Similar to how we capture superficial word-level statistics in § 3, we train a simple NLI model to make predictions based on bag-of-words features. We then identify samples which the model is uncertain about as the (new) testing examples. Our assumption is consistent with § 3 — simple models, limited by their capacity, can only rely on superficial word-level signals to make predictions. Therefore, samples not confidently answered by this model are potentially "genuiue" samples that require modeling the semantics of the corresponding input sentences in order to successfully infer their labels.

Specifically, we train a linear model, denoted by $u(X;\beta)$, and use the average of word embeddings as input:

$$u(X;\beta) = \text{softmax}(\mathbf{W}^{\beta}\frac{\sum_{i=1}^{l}\mathbf{x}_i}{l}),$$

where $\mathbf{W}^{\beta}$ is the weights of three-way classification. We merge the training, development, and testing split of an existing NLI dataset to optimize this model:

$$\widehat{\beta} = \arg\min_{\beta} l(\arg\max u(\mathbf{X};\beta), \mathbf{y}), \qquad (2)$$

where l is the cross-entropy loss. For every sample, we can compute an uncertainty score u':

$$u' = 1 - u(X;\widehat{\beta})_y,$$

where $u(X;\widehat{\beta})_y$ denotes the probability of label y. This uncertainty score measures how predictive the sample is by only considering word-level statistics. The smaller the uncertainty score, the easier the prediction can be made through superficial statistical regularities.

Finally, we can rearrange an existing NLI dataset and split it into training, development, and testing sets according to the uncertainty score. Let k_{te} and k_{dv} be the number of samples of the original testing and development set, the top k_{te} samples with highest uncertainty scores are selected as testing dataset, k_{dv} samples are randomly selected from the rest to be the development set, and the remaining ones are the training set. We use SNLI as an example and rearrange it. The resulting dataset is referred to as ReSNLI.

5 Experiments

5.1 Baselines

To verify the effectiveness of HEX, we plug it onto BERT and inspect the changes of the performances. Since BERT learns contextualized representations through masked language modeling on an extremely large corpus and achieves state-of-the-art performance for a wide range of tasks, we suggest that BERT is currently one of the most competitive NLI models and thus investigating the effect of HEX on BERT is compelling. We also report several other competitive NLI models as comparison. The baselines used in experiments are:

- InferSent (Conneau et al., 2017): A model consisting of sentence embedding, sequence encoder, composition layer, and top layer classifier. The top layer classifier is a MLP whose input is a concatenation of *premise* representation, *hypothesis* representation, the dot product of these two, and the absolute value of the difference of these two.

- ESIM (Enhanced Sequential Inference Model) (Chen et al., 2016): A method that uses local inference to model the relationship between *premise* and *hypothesis* after they are aligned locally.

- KIM (Knowledge-based Inference Model) (Chen et al., 2018b): This model enriches ESIM with external knowledge, including lexical semantic relation and whether two words are synonymy, antonymy, hypernymy, hyponymy, etc.

- BERT (Bidirectional Encoder Representations from Transformers) (Devlin et al., 2018)): We also fine-tune BERT, current state-of-the-art model for a wide range of tasks, on NLI datasets. Following common practice, we add a classification layer on the final hidden state corresponding to `[CLS]` to make NLI predictions.

5.2 Evaluation Settings

To evaluate models' robustness against superficial bag-of-words statistics, we run three different settings: swapping evaluation (Wang et al., 2019b), cross-domain evaluation (Talman and Chatzikyriakidis, 2018), and rearranging evaluation. The first two evaluations follow existing works to test models' resistance to swapping *premise* and *hypothesis* and models' ability to generalize across domains. Although providing interesting perspectives to evaluate the capability and limitation of NLI models, these evaluations are not specifically designed to inspect models' robustness to superficial word-level signals. We suggest that some confounding word-level statistics might still preserve when two sentences are swapped or evaluated on out-of-domain samples. Therefore, we use ReSNLI, an automatically rearranged SNLI dataset introduced in § 4, as a direct testbed to analyze models' inference ability beyond superficial bag-of-words signals. As shown in latter sections, ReSNLI is indeed more challenging than swapping evaluation and cross-domain evaluation, suggesting that more attention should be paid to improve models' true inference ability.

Swapping Evaluation As stated in Wang et al. (2019b), *contradiction* and *neutral* labels preserve if we swap *premise* and *hypothesis*, and a model that truly understands the semantic relationship between two sentences should be able to resist this swapping attack. To evaluate models' robustness against swapping attack, we swap sentences with *contradiction* or *neutral* labels in SNLI test set and investigate the performance drop. The smaller the gap, the more robust the model.

Cross-Domain Evaluation As Talman and Chatzikyriakidis (2018) recently reported, NLI models fail to generalize across different domains despite its high predictive performance when tested within one domain.

We follow the set-up of (Talman and Chatzikyriakidis, 2018) by testing the models across three data domains: SNLI (Bowman et al., 2015), MultiNLI (Williams et al., 2018), and SICK (Marelli et al., 2014). We plug in HEX and evaluate whether it can help improve the cross-domain performance. Specifically, we train on SNLI and MultiNLI datasets separately, and test the resulting models on SNLI, MultiNLI Matched (Multi-Mat), MultiNLI Mismatched (Multi-Mis), and SICK datasets to investigate cross-domain performance.

ReSNLI Evaluation We rearrange SNLI (Bowman et al., 2015) dataset so that the resulting test set is hard to be addressed by solely exploiting superficial bag-of-words signals. Models need to go beyond the association between word-level information and the NLI labels to perform well on this test set. Ideally, model evaluated with high accuracy on this test set will be less susceptible to the tendency in learning confounding bag-of-words statistics.

Table 1: Swapping evaluation of NLI models. Models trained on SNLI training set are evaluated on test set before and after swapping *premise* and *hypothesis* with *contradiction* or *neutral* labels. Bold indicates the best performing model.

Methods	Before Swap	After Swap
InferSent	82.82	73.82
ESIM	87.50	46.35
KIM	87.06	82.76
BERT	**89.55**	81.73
BERT + HEX	89.28	**82.84**

5.3 Experimental Results

Swapping Evaluation The results of the swapping evaluation are reported in Tab. 1. Performance drops significantly for all the models when they are tested on swapped datasets. Even the most powerful model BERT is observed to have an accuracy degradation by 7.82%. HEX applied to BERT successful reduces the gap to 6.44%, achieving best performance on the swapped SNLI test set. It suggests that discarding confounding signals can effective regularize the model to rely on semantic relationship between sentences, which can help resist swapping attack. Although the improvement is small in absolute terms, it is still impressive considering the fact that BERT already learns powerful contextualized representations.

Cross-Domain Evaluation As shown in Tab. 2, BERT with HEX achieves best performance on all cross-domain evaluation settings. The performance boost led by HEX is around $1\% \sim 2\%$ when tested on MultiNLI Mismatched (trained on SNLI) and SNLI (trained on MultiNLI), which is impressive considering the fact that BERT already achieves significantly superior results than other baselines in this cross-domain evaluation. The results suggest that HEX can effectively improve models' domain adaptation ability by discarding superficial word-level signals.

ReSNLI Evaluation The results of the rearranged evaluation are reported in Tab. 3. Comparing the results on SNLI and ReSNLI, we can see that the performance drops severely when the dataset is rearranged, indicating that ReSNLI is a challenging dataset to evaluate models' inference ability. The performance degradation on ReSNLI

Table 2: Cross-domain evaluation of NLI models. The first row are training data domains and the second row are testing domains. Bold indicates the best performing model.

Training	SNLI			MultiNLI	
Testing	MultiNLI-Mat	MultiNLI-Mis	SICK	SNLI	SICK
InferSent	54.01	55.01	33.85	53.01	54.02
ESIM	42.24	42.78	46.26	65.87	46.26
KIM	62.15	63.24	55.20	67.92	45.58
BERT	68.90	69.38	55.89	77.34	52.59
BERT + HEX	**70.21**	**70.30**	**56.06**	**77.69**	**54.65**

Table 3: Rearranging evaluation of NLI models. Models are evaluated on both SNLI and ReSNLI. Bold indicates the best performing model.

Methods	SNLI	ReSNLI
InferSent	82.82	33.77
ESIM	87.50	40.81
KIM	87.06	54.45
BERT	**89.55**	60.81
BERT + HEX	89.28	**61.94**

is even larger than that on swapped SNLI (Tab. 1), and that on cross-domain evaluation (Tab. 2), indicating that selecting samples least predictive by simple linear models as test set provide a complementary and more critical perspective to evaluate models' robustness that swapping evaluation and cross-domain evaluation do not touch.

Again, HEX improves BERT and achieves best performance on ReSNLI, which shows that discarding confounding signals is effective to regularize the model to learn more about how the human defines the NLI relationship. However, the gap between 89.28% (BERT on SNLI) and 61.94% (BERT on ReSNLI) is still quite large, meaning that understanding inference relationship between sentences is still far from being solved, and our rearranged datasets can serve as a challenging stress test to inspect models true inference ability beyond simple word-level signals.

Another advantage of our ReSNLI is that this data set can help evaluate the models at higher granularity with the uncertainty score calculated. For example, we split the ReSNLI testing data into twenty subsets according to the uncertainty score, from the easiest subset (ones with lowest uncertainty score) to the hardest subset (ones with highest uncertainty score) and evaluate the models with these partitions. The results are shown in Fig. 2.

The first message of Fig. 2 is that all these models tend to behave worse when the uncertainty score gets higher. However, the performances of these models differ from each other, indicating that these models have different understandings of the data. As the difficulty increases, the pioneering models (i.e., InferSent and ESIM) even predicts with a performance below the bar of random guess, while more powerful models (i.e., KIM, BERT, and BERT with HEX) can maintain the performance only slightly above random guess. HEX improves BERT in most of these partitions, and both of them outperform KIM with a clear advantage. On the other hand, we notice that KIM shows the highest prediction accuracy in the hardest partition, which suggests that the hardest samples may be better solved by explicit external knowledge, rather than pure data driven models. However, the detailed comparison between excessive-data driven approaches (BERT-like) and prior-knowledge driven approaches (KIM-like) is beyond the scope of this paper.

6 Case Study

Finally, we also present several samples in the ReSNLI dataset that we believe are challenging to classify if a model relies on the confounding signals. HEX can help correctly classify these examples, as shown in Tab. 4. Sample No. 1 is challenging because "empty" is usually associated with negation. As a result, BERT predicts *contradiction* based on this confounding signal. Sample No. 2 and No. 3 are confusing because "outside" is highly associated with *entailment* according to Gururangan et al. (2018). Thus BERT without HEX can easily be dominated by this spurious signal. Sample No. 4 and No. 5 contain a strong indicator of *contradiction* "no", thus misleading the predictions of BERT. By ignoring these confounding signals, HEX learns to rely on the semantic

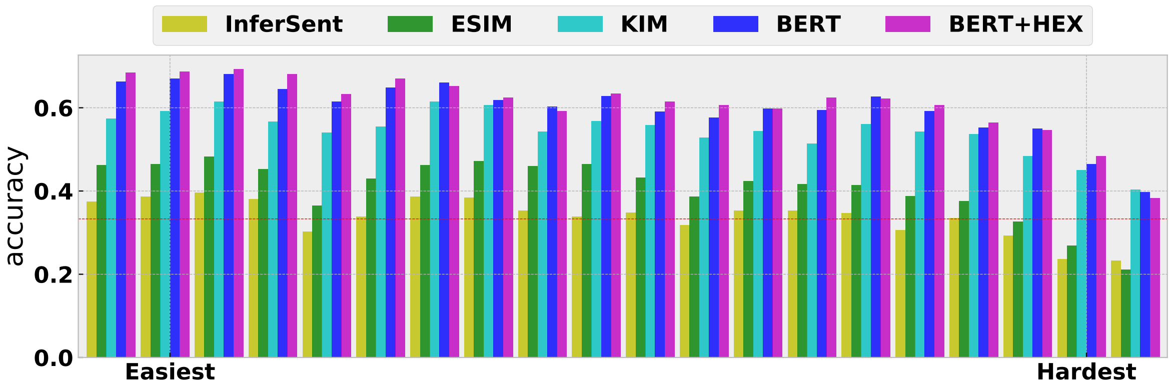

Figure 2: Detailed analysis of the results of ReSNLI when examined at higher granularity with uncertainty scores. The testing set is split into twenty partitions from the easiest to hardest according to the uncertainty score. The red horizontal line marks the performance of random guess.

relationship to makes correct prediction.

To investigate the limitation of HEX, we also study the examples correctly classified by the original model, but not by HEX. In SICK testing data set, we notice an extremely simple example with the *premise* “Dogs are racing on a track” and the *hypothesis* “There is no dog racing on a track” classified as *entailment* by HEX. This example is probably one of the simplest examples tested, yet HEX fails to classify it properly. We conjecture that this is because HEX regularizes the model to discard the simple confounding signals, thus, there might not be much information left for the model to make a prediction.

7 Related Work

NLI methods Since the introduction of large-scale NLI datasets (Bowman et al., 2015), many models have been proposed, most relying on LSTMs or Bidirectional LSTMs with various extensions. For example, Rocktäschel et al. (2016) extends LSTMs with a word-by-word neural attention mechanism. Wang and Jiang (2016) invented a match-LSTM to perform word-by-word matching of the two sentences. Liu et al. (2016a) introduced a pair of interdependent LSTMs, each modelling a sentence, which was further extended (Liu et al., 2016b,c). Cheng et al. (2016) replaced the single memory cell in an LSTM with a memory network. Sha et al. (2016) extended LSTM with a re-read mechanism to enhance the LSTM’s ability in learning the interconnection of the two sentences. Munkhdalai and Yu (2017); Choi et al. (2018) introduced variants of tree-structured recursive neural networks. Nie and Bansal (2017) utilized the LSTM with shortcut connections and fine-tuning of word embeddings Peters et al. (2018) introduced the deep contextualized word representations, which were trained with LSTM on large-scale text corpus. Kim et al. (2018) proposed a densely-connected co-attentive recurrent neural network that uses the concatenated information from any layer to all the subsequent layers. Chen et al. (2018a) introduced a vector-based multi-head attention as a generalized pooling method to enhance the performance of LSTM.

Further, we briefly summarize the more recent works regarding NLI: Tay et al. (2017) proposed a compare-propagate architecture which first compares the two text fragments and then propagates the aligned features to upper layers for representation learning. Shen et al. (2018) combined the benefit of soft attention and a newly proposed hard attention mechanism called reinforced sequence sampling (RSS). They further plugged this ReSA onto a source2token self-attention model and applied to NLI tasks. Tan et al. (2018) proposed a multiway attention network, which combines the information form four attention word-matching functions defined by four mathematical operations to build up the representation. Liu et al. (2018) introduced a stochastic answer network (SAN) for multi-step inference strategies for NLI. Unlike conventional methods that directly compute predictions given input sentence pairs, the SAN iteratively refines the predictions. Du et al. (2018) adopted Word-Pair-Dependency Triplets to improve model’s alignment and inference for the pair of sentences. Their method also helped improve

Table 4: Some examples that are incorrectly classified by the original model as a result of relying on superficial bag-of-words statistics, and HEX mitigates the problem.

No.	Premise/Hypothesis (keyword in italic)	Label	BERT	+HEX
1	There is no food at home and waiting for my son. The refrigerator is *empty*.	N	C	N
2	A man with a goatee wears a fur hat in the snow. A man *outside* is dressed in a coat.	N	E	N
3	A blond boy is sitting in a stroller holding a Korean language book. A blond boy is sitting in a stroller *outside*.	N	E	N
4	An ice cream truck employee is sitting on the side of the ice cream truck. An ice cream truck man is *not* doing his job.	E	C	E
5	People waiting for the subway. The subway has *not* arrived yet.	E	C	E

the model's interpretability. Guo et al. (2019) introduced an efficient Gaussian Transformer architecture that achieves higher predictive performance with fewer parameters in comparison to previous NLI models.

Techniques for Dicarding Confounding Signals Finally, we briefly introduce some related techniques that are introduced to overcoming the models' tendency to exploiting the confounding signals in general machine learning research: The most straightforward idea will be forcing the model to learn representations that are invariant to the confounding bias of data (e.g., Ganin et al., 2016; Wang et al., 2017, 2019c). Further, one can consider to apply data augmentation to dilute the confounding signals of data (e.g., Goyal et al., 2017; Jia and Liang, 2017; McCoy et al., 2019a; Liu et al., 2019).

8 Discussion

With detailed evaluation, we believe HEX can help in regularizing models to learn about how the human defines the inference relationship by discarding the confounding signals. Despite its impressive performance, several potential limitations need to be stated explicitly: (1) After HEX is plugged in, sometimes it takes more epochs for the model to converge. (2) HEX seems to be inferior in classifying samples whose inference relationship is determined directly by bag-of-words because HEX is designed to discard this information. (3) We also notice that HEX leads to poor performance when combined with KIM (Chen et al., 2018b). We conjecture this is partially due to that KIM is enriched by external knowledge of lexical semantic relations, including synonymy, antonymy, hypernymy, hyponymy, etc. Therefore, KIM may not depend on the distribution of words as much as other models do. However, the detailed analysis of how HEX should improve KIM is beyond the scope of this paper.

9 Conclusion

This paper is motivated by several recent observations showing that the high predictive performance of NLI models may not be the result of the models' abilities in "understanding" the inference relationship, but an outcome of the models' tendency in exploiting the confounding signals introduced during the construction of the datasets (Gururangan et al., 2018; Poliak et al., 2018; Glockner et al., 2018; Wang et al., 2019b; McCoy et al., 2019b).

We utilized a technique called HEX to discard these superficial signals and force the model to learn more high-level signals as how the human defines the inference relationship. Empirical results demonstrate the effectiveness of HEX.

We also proposed a simple technique to rearrange an existing dataset and select samples least predictive from word-level statistics to evaluate a models' robustness. We rearranged SNLI and showed that rearranging provides desired properties to evaluate a model's inference ability beyond superficial word-level signals.

References

Samuel R. Bowman, Gabor Angeli, Christopher Potts, and Christopher D. Manning. 2015. A large annotated corpus for learning natural language inference. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (EMNLP)*.

Qian Chen, Zhen-Hua Ling, and Xiaodan Zhu. 2018a. Enhancing sentence embedding with generalized pooling. In *COLING*.

Qian Chen, Xiaodan Zhu, Zhen-Hua Ling, Diana Inkpen, and Si Wei. 2018b. Neural natural language inference models enhanced with external knowledge. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*.

Qian Chen, Xiaodan Zhu, Zhenhua Ling, Si Wei, and Hui Jiang. 2016. Enhancing and combining sequential and tree lstm for natural language inference. *arXiv preprint arXiv:1609.06038*.

Jianpeng Cheng, Li Dong, and Mirella Lapata. 2016. Long short-term memory-networks for machine reading. In *EMNLP*.

Jihun Choi, Kang Min Yoo, and Sang-goo Lee. 2018. Learning to compose task-specific tree structures. In *Proceedings of the 2018 Association for the Advancement of Artificial Intelligence (AAAI). and the 7th International Joint Conference on Natural Language Processing (ACL-IJCNLP)*.

Alexis Conneau, Douwe Kiela, Holger Schwenk, Loïc Barrault, and Antoine Bordes. 2017. Supervised learning of universal sentence representations from natural language inference data. In *EMNLP*.

Ido Dagan, Oren Glickman, and Bernardo Magnini. 2005. The pascal recognising textual entailment challenge. In *Machine Learning Challenges Workshop*, pages 177–190. Springer.

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

Qianlong Du, Chengqing Zong, and Keh-Yih Su. 2018. Adopting the word-pair-dependency-triplets with individual comparison for natural language inference. In *Proceedings of the 27th International Conference on Computational Linguistics*.

Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. 2016. Domain-adversarial training of neural networks. *The Journal of Machine Learning Research*, 17(1):2096–2030.

Max Glockner, Vered Shwartz, and Yoav Goldberg. 2018. Breaking nli systems with sentences that require simple lexical inferences. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*.

Yash Goyal, Tejas Khot, Douglas Summers-Stay, Dhruv Batra, and Devi Parikh. 2017. Making the v in vqa matter: Elevating the role of image understanding in visual question answering. In *CVPR*.

Maosheng Guo, Yu Zhang, and Ting Liu. 2019. Gaussian transformer: a lightweight approach for natural language inference. In *Proceedings of AAAI*.

Suchin Gururangan, Swabha Swayamdipta, Omer Levy, Roy Schwartz, Samuel R. Bowman, and Noah A. Smith. 2018. Annotation artifacts in natural language inference data. In *NAACL-HLT*.

R. Jia and P. Liang. 2017. Adversarial examples for evaluating reading comprehension systems. In *Empirical Methods in Natural Language Processing (EMNLP)*.

Seonhoon Kim, Jin-Hyuk Hong, Inho Kang, and Nojun Kwak. 2018. Semantic sentence matching with densely-connected recurrent and co-attentive information. *arXiv preprint arXiv:1805.11360*.

N. F. Liu, R. Schwartz, and N. A. Smith. 2019. Inoculation by fine-tuning: A method for analyzing challenge datasets. In *North American Association for Computational Linguistics (NAACL)*.

Pengfei Liu, Xipeng Qiu, Jifan Chen, and Xuanjing Huang. 2016a. Deep fusion lstms for text semantic matching. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, volume 1, pages 1034–1043.

Pengfei Liu, Xipeng Qiu, Yaqian Zhou, Jifan Chen, and Xuanjing Huang. 2016b. Modelling interaction of sentence pair with coupled-lstms. In *EMNLP*.

Xiaodong Liu, Kevin Duh, and Jianfeng Gao. 2018. Stochastic answer networks for natural language inference. *arXiv preprint arXiv:1804.07888*.

Yang Liu, Chengjie Sun, Lei Lin, and Xiaolong Wang. 2016c. Learning natural language inference using bidirectional lstm model and inner-attention. *arXiv preprint arXiv:1605.09090*.

Bill MacCartney and Christopher D Manning. 2009. *Natural language inference*. Citeseer.

Marco Marelli, Stefano Menini, Marco Baroni, Luisa Bentivogli, Raffaella Bernardi, Roberto Zamparelli, et al. 2014. A sick cure for the evaluation of compositional distributional semantic models. In *LREC*, pages 216–223.

R. T. McCoy, E. Pavlick, and T. Linzen. 2019a. Right for the wrong reasons: Diagnosing syntactic heuristics in natural language inference. *arXiv preprint arXiv:1902.01007*.

R Thomas McCoy, Ellie Pavlick, and Tal Linzen. 2019b. Right for the wrong reasons: Diagnosing syntactic heuristics in natural language inference. *arXiv preprint arXiv:1902.01007*.

Tsendsuren Munkhdalai and Hong Yu. 2017. Neural tree indexers for text understanding. In *Proceedings of the conference. Association for Computational Linguistics. Meeting*, volume 1.

Aakanksha Naik, Abhilasha Ravichander, Norman Sadeh, Carolyn Penstein Rosé, and Graham Neubig. 2018. Stress test evaluation for natural language inference. In *COLING*.

Yixin Nie and Mohit Bansal. 2017. Shortcut-stacked sentence encoders for multi-domain inference. In *RepEval@EMNLP*.

Matthew E. Peters, Mark Neumann, Mohit Iyyer, Matt Gardner, Christopher Clark, Kenton Lee, and Luke Zettlemoyer. 2018. Deep contextualized word representations. In *NAACL-HLT*.

Adam Poliak, Jason Naradowsky, Aparajita Haldar, Rachel Rudinger, and Benjamin Van Durme. 2018. Hypothesis only baselines in natural language inference. In *NAACL-HLT*.

Tim Rocktäschel, Edward Grefenstette, Karl Moritz Hermann, Tomas Kocisky, and Phil Blunsom. 2016. Reasoning about entailment with neural attention. In *International Conference on Learning Representations (ICLR)*.

Lei Sha, Baobao Chang, Zhifang Sui, and Sujian Li. 2016. Reading and thinking: Re-read lstm unit for textual entailment recognition. In *Proceedings of COLING 2016, the 26th International Conference on Computational Linguistics: Technical Papers*, pages 2870–2879.

Tao Shen, Tianyi Zhou, Guodong Long, Jing Jiang, Sen Wang, and Chengqi Zhang. 2018. Reinforced self-attention network: a hybrid of hard and soft attention for sequence modeling. *arXiv preprint arXiv:1801.10296*.

Aarne Talman and Stergios Chatzikyriakidis. 2018. Testing the generalization power of neural network models across nli benchmarks. *arXiv preprint arXiv:1810.09774*.

Chuanqi Tan, Furu Wei, Wenhui Wang, Weifeng Lv, and Ming Zhou. 2018. Multiway attention networks for modeling sentence pairs. In *IJCAI*, pages 4411–4417.

Yi Tay, Luu Anh Tuan, and Siu Cheung Hui. 2017. A compare-propagate architecture with alignment factorization for natural language inference. *arXiv preprint arXiv:1801.00102*.

Haohan Wang, Zexue He, Zachary L. Lipton, and Eric P. Xing. 2019a. Learning robust representations by projecting superficial statistics out. In *International Conference on Learning Representations*.

Haohan Wang, Aaksha Meghawat, Louis-Philippe Morency, and Eric P Xing. 2017. Select-additive learning: Improving cross-individual generalization in multimodal sentiment analysis. In *IEEE International Conference on Multimedia and Expo*.

Haohan Wang, Da Sun, and Eric P Xing. 2019b. What if we simply swap the two text fragments? a straightforward yet effective way to test the robustness of methods to confounding signals in nature language inference tasks. In *Proceedings of AAAI*.

Haohan Wang, Zhenglin Wu, and Eric P Xing. 2019c. Removing confounding factors associated weights in deep neural networks improves the prediction accuracy for healthcare applications. In *Proceedings of PSB*.

Shuohang Wang and Jing Jiang. 2016. Learning natural language inference with lstm. In *NAACL-HLT*.

Adina Williams, Nikita Nangia, and Samuel R. Bowman. 2018. A broad-coverage challenge corpus for sentence understanding through inference. In *NAACL-HLT*.