The Curse of Performance Instability in Analysis Datasets: Consequences, Source, and Suggestions

Xiang Zhou **Yixin Nie** **Hao Tan** **Mohit Bansal**
Department of Computer Science
University of North Carolina at Chapel Hill
{xzh, yixin1, haotan, mbansal}@cs.unc.edu

Abstract

We find that the performance of state-of-the-art models on Natural Language Inference (NLI) and Reading Comprehension (RC) analysis/stress sets can be highly unstable. This raises three questions: (1) How will the instability affect the reliability of the conclusions drawn based on these analysis sets? (2) Where does this instability come from? (3) How should we handle this instability and what are some potential solutions? For the first question, we conduct a thorough empirical study over analysis sets and find that in addition to the unstable final performance, the instability exists all along the training curve. We also observe lower-than-expected correlations between the analysis validation set and standard validation set, questioning the effectiveness of the current model-selection routine. Next, to answer the second question, we give both theoretical explanations and empirical evidence regarding the source of the instability, demonstrating that the instability mainly comes from high inter-example correlations within analysis sets. Finally, for the third question, we discuss an initial attempt to mitigate the instability and suggest guidelines for future work such as reporting the decomposed variance for more interpretable results and fair comparison across models.[1]

1 Introduction

Neural network models have significantly pushed forward performances on natural language processing benchmarks with the development of large-scale language model pre-training (Peters et al., 2018; Radford et al., 2018; Devlin et al., 2019; Radford et al., 2019; Liu et al., 2019b). For example, on two semantically challenging tasks, Natural Language Inference (NLI) and Reading Comprehension (RC), the state-of-the-art results have reached or even surpassed the estimated human performance on certain benchmark datasets (Wang et al., 2019; Rajpurkar et al., 2016a, 2018). These astounding improvements, in turn, motivate a new trend of research to analyze what language understanding and reasoning skills are actually achieved, versus what is still missing within these current models. Following this trend, numerous analysis approaches have been proposed to examine models' ability to capture different linguistic phenomena (e.g., named entities, syntax, lexical inference, etc.). Those studies are often conducted in 3 steps: (1) proposing assumptions about a certain ability of the model; (2) building analysis datasets by automatic generation or crowd-sourcing; (3) concluding models' ability using results on these analysis datasets.

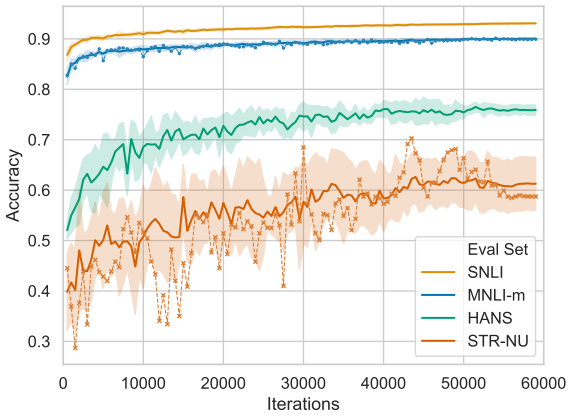

Figure 1: The trajectories of BERT performance on SNLI, MNLI-m, HANS (McCoy et al., 2019b), and the Numerical subcategory of the Stress Test dataset (Naik et al., 2018a) (from the topmost line to the bottom, respectively). The solid lines represent the means of ten runs and the shadow area indicates a distance within a standard deviation from the means. The two dashed lines show the trajectories of one single run for MNLI-m and Numerical Stress Test using the same model.

Past analysis studies have led to many key dis-

[1]Our code is publicly available at: https://github.com/owenzx/InstabilityAnalysis

Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, pages 8215–8228,
November 16–20, 2020. ©2020 Association for Computational Linguistics

coveries in NLP models, such as over-stability (Jia and Liang, 2017), surface pattern overfitting (Gururangan et al., 2018), but recently McCoy et al. (2019a) found that the results of different runs of BERT NLI models have large non-negligible variances on the HANS (McCoy et al., 2019b) analysis datasets, contrasting sharply with their stable results on standard validation set across multiple seeds. This finding raises concerns regarding the reliability of individual results reported on those datasets, the conclusions made upon these results, and lack of reproducibility (Makel et al., 2012). Thus, to help consolidate further developments, we conduct a deep investigation on model instability, showing how unstable the results are, and how such instability compromises the feedback loop between model analysis and model development.

We start our investigation from a thorough empirical study of several representative models on both NLI and RC. Overall, we observe four worrisome observations in our experiments: (1) The final results of the same model with different random seeds on several analysis sets are of significantly **high variance**. The largest variance is more than 27 times of that for standard development set; (2) These large instabilities on certain datasets is **model-agnostic**. Certain datasets have unstable results across different models; (3) The instability not only occurs at the final performance but exists **all along training trajectory**, as shown in Fig. 1; (4) The results of the same model on analysis sets and on the standard development set have **low correlation**, making it hard to draw any constructive conclusion and questioning the effectiveness of the standard model-selection routine.

Next, in order to grasp a better understanding of this instability issue, we explore theoretical explanations behind this instability. Through our theoretical analysis and empirical demonstration, we show that inter-examples correlation within the dataset is the dominating factor causing this performance instability. Specifically, the variance of model accuracy on the entire analysis set can be decomposed into two terms: (1) the sum of single-data variance (the variance caused by individual prediction randomness on each example), and (2) the sum of inter-data covariance (caused by the correlation between different predictions). To understand the latter term better, consider the following case: if there are many examples correlated with each other in the evaluation set, then the change of prediction on one example will influence predictions on all the correlated examples, causing high variances in final accuracy. We estimate these two terms with multiple runs of experiments and show that inter-data covariance contributes significantly more than single-data variance to final accuracy variance, indicating its major role in the cause of instability.

Finally, in order for the continuous progress of the community to be built upon trustworthy and interpretable results, we provide initial suggestions on how to perceive the implication of this instability issue and how we should potentially handle it. For this, we encourage future research to: (1) when reporting means and variance over multiple runs, also report two decomposed variance terms (i.e., sum of single data variance and sum of inter-data covariance) for more interpretable results and fair comparison across models; (2) focus on designing models with better inductive and structural biases, and datasets with higher linguistic diversity.

Our contribution is 3-fold. First, we provide a thorough empirical study of the instability issue in models' performance on analysis datasets. Second, we demonstrate theoretically and empirically that the performance variance is attributed mostly to inter-example correlations. Finally, we provide suggestions on how to deal with instability, including reporting the decomposed variance for more interpretable evaluation and better comparison.

2 Related Work

NLI and RC Analysis. Many analysis works have been conducted to study what the models are actually capturing alongside recent improvements on NLI and RC benchmark scores. In NLI, some analyses target word/phrase level lexical/semantic inference (Glockner et al., 2018; Shwartz and Dagan, 2018; Carmona et al., 2018), some are more syntactic-related (McCoy et al., 2019b; Nie et al., 2019; Geiger et al., 2019), some also involved logical-related study (Minervini and Riedel, 2018; Wang et al., 2019). Naik et al. (2018a) proposed a suite of analysis sets covering different linguistic phenomena. In RC, adversarial style analysis is used to test the robustness of the models (Jia and Liang, 2017). Most of the work follows the style of Carmona et al. (2018) to diagnose/analyze models' behavior on pre-designed analysis sets. In this paper, we analyze NLI and RC models from a broader perspective by inspecting models' performance across different analysis sets, and their

inter-dataset and intra-dataset relationships.

Dataset-Related Analysis. Another line of works study the meta-issues of the dataset. The most well-known one is the analysis of undesirable bias. In VQA datasets, unimodal biases were found, compromising their authority on multimodality evaluation (Jabri et al., 2016; Goyal et al., 2017). In RC, Kaushik and Lipton (2018) found that passage-only models can achieve decent accuracy. In NLI, hypothesis bias was also found in SNLI and MultiNLI (Tsuchiya, 2018; Gururangan et al., 2018). These findings revealed the spurious shortcuts in the dataset and their harmful effects on trained models. To mitigate these problems, Liu et al. (2019a) introduced a systematic task-agnostic method to analyze datasets. Rozen et al. (2019) further explain how to improve challenging datasets and why diversity matters. Geva et al. (2019) suggest that the training and test data should be from exclusive annotators to avoid annotator bias. Our work is complementary to those analyses.

Robustifying NLI and RC Models. Recently, a number of works have been proposed to directly improve the performance on the analysis datasets both for NLI through model ensemble (Clark et al., 2019; He et al., 2019), novel training mechanisms (Pang et al., 2019; Yaghoobzadeh et al., 2019), adversarial data augmentation (Nie et al., 2020), enhancing word representations (Moosavi et al., 2019), and for RC through different training objectives (Yeh and Chen, 2019; Lewis and Fan, 2019). While improvements have been made on certain analysis datasets, the stability of the results is not examined. As explained in this paper, we highly recommend those result variances be scrutinized in future work for fidelity considerations.

Instability in Performance. Performance instability has already been recognized as an important issue in deep reinforcement learning (Irpan, 2018) and active learning (Bloodgood and Grothendieck, 2013). However, supervised learning is presumably stable especially with fixed datasets and labels. This assumption is challenged by some analyses recently. McCoy et al. (2019a) show high variances in NLI-models performance on the analysis dataset. Phang et al. (2018) found high variances in fine-tuning pre-trained models in several NLP tasks on the GLUE Benchmark. Reimers and Gurevych (2017, 2018) state that conclusions based on single run performance may not be reliable for machine learning approaches. Weber et al. (2018) found that the model's ability to generalize beyond the training distribution depends greatly on the random seed. Dodge et al. (2020) showed weight initialization and training data order both contribute to the randomness in BERT performance. In our work, we present a comprehensive explanation and analysis of the instability of neural models on analysis datasets and give general guidance for future work.

3 The Curse of Instability

3.1 Tasks and Datasets

In this work, we target our experiments on NLI and RC for two reasons: 1) their straightforwardness for both automatic evaluation and human understanding, and 2) their wide acceptance of being benchmarks evaluating natural language understanding.

For NLI, we use SNLI (Bowman et al., 2015) and MNLI (Williams et al., 2018) as the main standard datasets and use HANS (McCoy et al., 2019b), SNLI-hard (Gururangan et al., 2018), BREAK-NLI (Glockner et al., 2018), Stress Test (Naik et al., 2018a), SICK (Marelli et al., 2014), EQUATE (Ravichander et al., 2019) as our auxiliary analysis sets. Note that the Stress Test contains 6 subsets (denoted as 'STR-X') targeting different linguistic categories. We also splite the EQUATE dataset to two subsets (denoted as 'EQU-NAT/SYN') based on whether the example are from natural real-world sources or are controlled synthetic tests. For RC, we use SQuAD1.1 (Rajpurkar et al., 2016b) as the main standard dataset and use AdvSQuAD (Jia and Liang, 2017) as the analysis set. All the datasets we use in this paper are English. Detailed descriptions of the datasets are in Appendix.

3.2 Models

Since BERT (Devlin et al., 2019) achieves state-of-the-art results on several NLP tasks, the pretraining-then-finetuning framework has been widely used. To keep our analysis aligned with recent progress, we focused our experiments on this framework. Specifically, in our study, we used the two most typical choices: BERT (Devlin et al., 2019) and XLNet (Yang et al., 2019).[2] Moreover, for NLI, we additionally use RoBERTa (Liu et al., 2019b)

[2]For all the transformer models, we use the implementation in https://github.com/huggingface/transformers. BERT-B, BERT-L stands for BERT-base and BERT-large, respectively. The same naming rule applies to other transformer models.

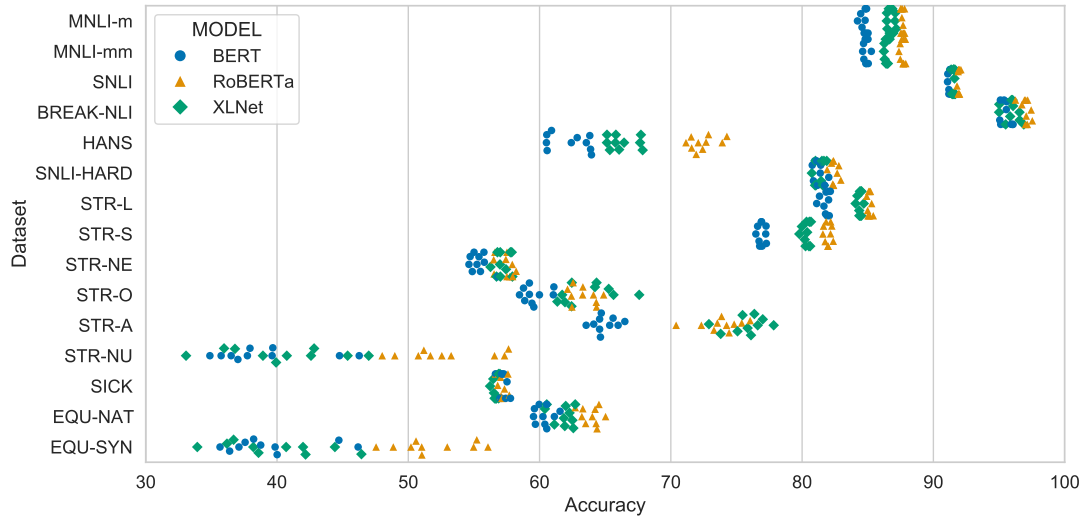

Figure 2: The results of BERT, RoBERTa, and XLNet on all datasets with 10 different random seeds. Large variance can be seen at certain analysis datasets (e.g. STR-NU, HANS, etc.) while results on standard validation sets are always stable.

and ESIM (Chen et al., 2017) in our experiments. RoBERTa is almost the same as BERT except that it has been trained on 10 times more data during the pre-training phrase to be more robust. ESIM is the most representative pre-BERT model for sequence matching problem and we used an ELMo-enhanced-version (Peters et al., 2018).[3] All the models and training details are in Appendix.

3.3 What are the Concerns?

Instability in Final Performance. Models' final results often serve as a vital measurement for comparative study. Thus, we start with the question: "How unstable are the final results?" To measure the instability, we train every model 10 times with different random seeds. Then, we evaluate the performances of all the final checkpoints on each NLI dataset and compute their standard deviations. As shown in Fig. 2, the results of different runs for BERT, RoBERTa, and XLNet are highly stable on MNLI-m, MNLI-mm, and SNLI, indicating that model performance on standard validation datasets regardless of domain consistency[4] are fairly stable. This stability also holds on some analysis sets, especially on SNLI-hard, which is a strict subset of the SNLI validation set. On the contrary, there are noticeable high variances on some analysis sets. The most significant ones are on STR-NU and HANS where points are sparsely scattered, with a 10-point gap between the highest and the lowest number for STR-NU and a 4-point gap for HANS.

Model-Agnostic Instability. Next, we check if the instability issue is model-agnostic. For a fair comparison, as the different sizes of the datasets will influence the magnitude of the instability, we normalize the standard deviation on different datasets by multiplying the square root of the size of the dataset[5] and focus on the relative scale compared to the results on the MNLI-m development set, i.e., $\frac{STD(dataset)}{STD(MNLI-m)}\sqrt{\frac{SIZE(dataset)}{SIZE(MNLI-m)}}$. The results for all the models are shown in Table 1 (the original means and standard deviations are in Appendix). From Table 1, we can see that the instability phenomenon is consistent across all the models. Regardless of the model choice, some of the analysis datasets (e.g., HANS, STR-O, STR-N) are significantly more unstable (with standard deviation 27 times larger in the extreme case) than the standard evaluation datasets. Similarly, for RC, the normalized deviation of model F1 results on SQuAD almost doubled when evaluated on AddSent, as shown in Table 2 (the original means and standard deviations are in Appendix).

Fluctuation in Training Trajectory. Intuitively, the inconsistency and instability in the final performance of different runs can be caused by the

[3]For ESIM, we use the implementation in AllenNLP (Gardner et al., 2018).

[4]Here SNLI and MNLI-m share the same domain as the training set while MNLI-mm is from different domains.

[5]This normalization factor assumes that every prediction is independent of each other.

Model	Standard Datasets			Analysis Sets											
	MNLI-m	MNLI-mm	SNLI	BREAK-NLI	HANS	SNLI-hard	STR-L	STR-S	STR-NE	STR-O	STR-A	STR-NU	SICK	EQU-NAT	EQU-SYN
ESIM	1.00	0.57	0.73	3.84	0.82	0.73	0.77	0.73	3.57	**4.63**	2.58	2.79	1.47	1.19	2.70
ESIM+ELMo	1.00	2.00	1.50	11.5	4.55	2.48	3.10	2.20	7.50	**15.5**	6.38	8.36	2.28	2.36	8.45
BERT-B	1.00	0.83	0.48	1.43	10.95	0.95	1.39	1.04	2.70	3.70	1.46	**13.65**	1.48	1.03	13.17
RoBERTa-B	1.00	1.46	0.64	2.82	15.42	1.47	1.27	2.17	5.45	8.45	5.55	**25.75**	2.91	2.29	22.68
XLNet-B	1.00	0.48	0.37	2.03	6.60	0.75	0.59	0.92	1.96	7.19	2.07	13.33	0.82	1.15	**13.33**
BERT-L	1.00	1.13	0.56	2.86	18.47	1.37	1.31	2.63	9.19	10.13	2.39	**21.88**	1.71	1.41	20.36
RoBERTa-L	1.00	0.88	0.69	1.03	10.27	1.01	1.12	1.20	12.13	10.13	4.51	**27.38**	1.71	1.21	22.36
XLNet-L	1.00	0.90	0.69	1.06	10.67	0.85	0.89	1.45	16.21	11.84	4.26	15.93	1.50	1.31	**19.93**

Table 1: Relatively normalized deviations of the results on MNLI-m for all models. The highest deviations are in bold and the second highest deviations are underlined for each individual model.

Model	Standard Dataset	Analysis Sets	
	SQuAD	AddSent	AddOneSent
BERT-B	1.00	**2.61**	1.58
XLNet-B	1.00	**1.78**	1.00

Table 2: Relatively normalized deviations of the results on SQuAD dev set for both BERT-B and XLNet-B.

randomness in initialization and stochasticity in training dynamics. To see how much these factors can contribute to the inconsistency in the final performance, we keep track of the results on different evaluation sets along the training process and compare their training trajectories. We choose HANS and STR-NU as our example unstable analysis datasets because their variances in final performance are the largest, and we choose SNLI and MNLI-m for standard validation set comparison. As shown in Fig. 1, the training curve on MNLI and SNLI (the top two lines) is highly stable, while there are significant fluctuations in the HANS and STR-NU trajectories (bottom two lines). Besides the mean and standard deviation over multiple runs, we also show the accuracy of one run as the bottom dashed line in Fig. 1. We find that two adjacent checkpoints can have a dramatically large performance gap on STR-NU. Such fluctuation is very likely to be one of the reasons for the instability in the final performance and might give rise to untrustworthy conclusions drawn from the final results.

Low Correlation between Datasets. The typical routine for neural network model selection requires practitioners to choose the model or checkpoint hinged on the observation of models' performance on the validation set. The routine was followed in all previous NLI analysis studies where models were chosen by the performance on standard validation set and tested on analysis sets. An important assumption behind this routine is that the performance on the validation set should be correlated with the models' general ability. However, as

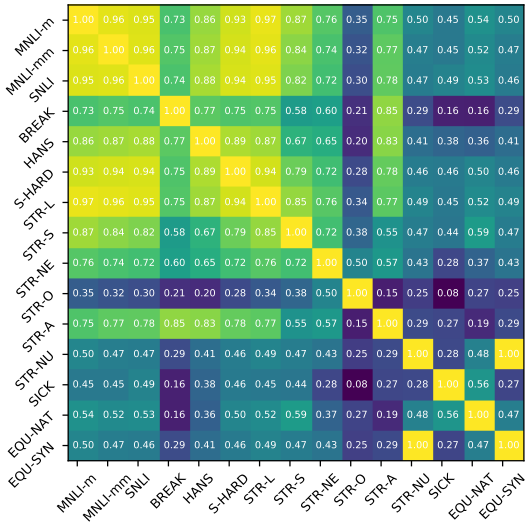

Figure 3: Spearman's correlations for different datasets showing the low correlation between standard datasets (i.e., MNLI-m, MNLI-mm, and SNLI) and all the other analysis datasets.

shown in Fig. 1, the striking difference between the wildly fluctuated training curves for analysis sets and the smooth curves for the standard validation set questions the validity of this assumption.

Therefore, to check the effectiveness of model selection under these instabilities, we checked the correlation for the performance on different datasets during training. For dataset $\mathcal{D}^i$, we use $a^i_{t,s}$ to denote the accuracy of the checkpoint at t-th time step and trained with the seed $s \in S$, where S is the set of all seeds. We calculate the correlation $\text{Corr}_{i,j}$ between datasets $\mathcal{D}^i$ and $\mathcal{D}^j$ by:

$$\text{Corr}_{i,j} = \frac{1}{|S|} \sum_{s \in S} \text{Spearman}\left[(a^i_{t,s})^T_{t=1}, (a^j_{t,s})^T_{t=1}\right]$$

where T is the number of checkpoints.

The correlations between different NLI datasets are shown in Fig. 3. We can observe high correlation (> 0.95) among standard validation datasets (e.g. MNLI-m, MNLI-mm, SNLI) but low correlations between other dataset pairs, especially when pairing STR-O or STR-NU with MNLI or

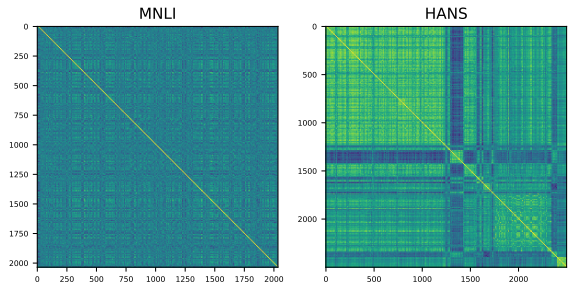

Figure 4: The two heatmaps of inter-example correlations matrices for both MNLI and HANS. Each point in the heatmap represents the Spearman's correlation between the predictions of an example-pair.

SNLI. While these low correlations between standard evaluation sets and analysis sets can bring useful insights for analysis, this also indicates that: 1) performance on the standard validation set is not representative enough for certain analysis set performances; 2) comparison/conclusions drawn from analysis datasets' results from model selection on standard evaluation sets may be unreliable.

4 Tracking Instability

Before answering the question how to handle these instabilities, we first seek the source of the instability to get a better understanding of the issue. We start with the intuition that high variance could be the result of high inter-example correlation within the dataset, and then provide hints from experimental observations. Next, we show theoretical evidence to formalize our claim. Finally, we conclude that the major source of variance is the inter-example correlations based on empirical results.

4.1 Inter-Example Correlations

Presumably, the wild fluctuation in the training trajectory on different datasets might come from two potential sources. Firstly, the individual prediction of each example may be highly unstable so that the prediction is constantly changing. Secondly, there might be strong inter-example correlations in the datasets such that a large proportion of predictions are more likely to change simultaneously, thus causing large instability. Here we show that the second reason, i.e., the strong inter-example prediction correlation is the major factor.

We examine the correlation between different example prediction pairs during the training process. In Fig. 4, we calculated the inter-example Spearman's correlation on MNLI and HANS. Fig. 4 shows a clear difference between the inter-example correlation in stable (MNLI) datasets versus unstable (HANS) datasets. For stable datasets (MNLI), the correlations between the predictions of examples are uniformly low, while for unstable datasets (HANS), there exist clear groups of examples with very strong inter-correlation between their predictions. This observation suggests that those groups could be a major source of instability if they contain samples with frequently changing predictions.

4.2 Variance Decomposition

Next, we provide theoretical support to show how the high inter-example correlation contributes to the large variance in final accuracy. Later, we will also demonstrate that it is the major source of the large variance. Suppose dataset $\mathcal{D}$ contains examples $\{x_i, y_i\}_{i=1}^N$, where N is the number of data points in the dataset, x_i and y_i are the inputs and labels, respectively. We use a random variable C_i to denote whether model M predicts the i-th example correctly: $C_i = \mathbb{1}[y_i = M(x_i)]$. We ignore the model symbol M in our later notations for simplicity. The accuracy Acc of model M is another random variable, which equals to the average over $\{C_i\}$, w.r.t. different model weights (i.e., caused by different random seeds in our experiments): $Acc = \frac{1}{N}\sum_i C_i$. We then decompose the variance of the accuracy $\mathrm{Var}(Acc)$ into the sum of data variances $\mathrm{Var}(C_i)$, and the sum of inter-data covariances $\mathrm{Cov}(C_i, C_j)$:

$$\begin{aligned}\mathrm{Var}(Acc) &= \frac{1}{N^2}\mathrm{Cov}\left(\sum_{i=1}^{N} C_i,\ \sum_{j=1}^{N} C_j\right) \\ &= \frac{1}{N^2}\sum_{i=1}^{N}\sum_{j=1}^{N}\mathrm{Cov}\left(C_i,\ C_j\right) \\ &= \frac{1}{N^2}\sum_{i=1}^{N}\mathrm{Var}(C_i) + \frac{2}{N^2}\sum_{i<j}\mathrm{Cov}(C_i, C_j)\end{aligned} \tag{1}$$

Here, the first term $\frac{1}{N^2}\sum \mathrm{Var}(C_i)$ means the instability caused by the randomness in individual example prediction and the second term $\frac{2}{N^2}\sum_{i<j}\mathrm{Cov}(C_i, C_j)$ means the instability caused by the covariance of the prediction between different examples. The latter covariance term is highly related to the inter-example correlation.

Finally, to demonstrate that the inter-example correlation is the major source of high variance, we calculate the total variance, the independent variance (the 1st term in Eq. 1), and the covariance (the

Statistics	Standard Dataset			Analysis Dataset											
	MNLI-m	**MNLI-mm**	**SNLI**	**BREAK**	**HANS**	**SNLI-hard**	**STR-L**	**STR-S**	**STR-NE**	**STR-O**	**STR-A**	**STR-NU**	**SICK**	**EQU-NAT**	**EQU-SYN**
$\sqrt{\text{Total Var}}$	0.24	0.20	0.11	0.38	1.51	0.40	0.34	0.28	0.65	0.90	0.89	3.76	0.35	0.66	3.47
$\sqrt{\text{Idp Var}}$	0.18	0.18	0.13	0.12	0.10	0.30	0.17	0.22	0.17	0.19	0.56	0.33	0.17	0.59	0.31
$\sqrt{\lvert\text{Cov}\rvert}$	0.16	0.09	0.06	0.36	1.51	0.27	0.28	0.15	0.63	0.88	0.69	3.74	0.31	0.31	3.45

Table 3: The square roots of total variance (Total Var), independent variance (Idp Var), and the absolute covariance (|Cov|) of BERT model on different NLI datasets. Square root is applied to map variances and covariances to a normal range. Analysis datasets have much higher covariance than standard datasets.

Statistics	Standard Dataset	Analysis Dataset	
	SQuAD	**AddSent**	**AddOneSent**
$\sqrt{\text{Total Var}}$	0.13	0.57	0.48
$\sqrt{\text{Idp Var}}$	0.15	0.33	0.44
$\sqrt{\lvert\text{Cov}\rvert}$	0.09	0.43	0.13

Table 4: The square roots of total variance (Total Var), independent variance (Idp Var), and absolute covariance (|Cov|) of BERT model on different RC datasets.

Premise:	Though the author encouraged the lawyer, the tourist waited.
Hypothesis:	The author encouraged the lawyer.
Label:	entailment
Premise:	The lawyer thought that the senators supported the manager.
Hypothesis:	The senators supported the manager.
Label:	non-entailment

Table 5: A highly-correlated example pair in the HANS dataset with the BERT model. This example pair have the largest covariance (0.278) among all the pairs.

2nd term in Eq. 1) on every dataset in Table 3. In contrast to similar averages of the independent variance on standard and analysis datasets, we found a large gap between the averages of covariances on different datasets. This different trend of total variance and independent variance proves that the inter-example correlation is the major reason for the difference of variance on the analysis datasets.

4.3 Highly-Correlated Cases

From these analyses, we can see that one major reason behind the high variance in certain analysis datasets is high inter-example correlation. Following this direction, the next question is why these highly-correlated example-pairs are more likely to appear in analysis datasets. From Table 1, we can find that the largest variance happens in HANS, several subsets of STR, and EQU-SYN. On the other hand, while datasets like SNLI-hard and EQU-NAT are also analysis datasets, their variance is much smaller than the former ones. One crucial difference among the high-variance datasets is that they are usually created with the help of synthetic rules.

This way of well-controlled synthetic-rule based construction can effectively target certain linguistic phenomena in the dataset, but they may also cause many examples to share similar lexicon usage. One example from the HANS dataset is shown in Table 5, and another similar example for RC is also shown in Appendix. These similarities in syntax and lexicon are very likely to cause the prediction in these two examples to be highly-correlated. Another evidence can also be seen from Figure 4, where we can see clear boundaries of blocks of high-correlation examples in the right sub-figure (for HANS dataset). Since the examples in HANS are ordered by its templates, examples in the same block are created using the same template. Hence, the block patterns in the figure also show how synthetic rules may cause predictions to be more correlated with each other.

In conclusion, since analysis datasets are sometimes created using pre-specified linguistic patterns/properties and investigation phenomena in mind, the distributions of analysis datasets are less diverse than the distributions of standard datasets. The difficulty of the dataset and the lack of diversity can lead to highly-correlated predictions and high instability in models' final performances.

5 Implications, Suggestions, and Discussion

So far, we have demonstrated how severe this instability issue is and how the instability can be traced back to the high correlation between predictions of certain example clusters. Now based on all the previous analysis results, we discuss potential ways of how to deal with this instability issue.

We first want to point out that this instability issue is not a simple problem that can be solved by trivial modifications of the dataset, model, or training algorithm. Here, below we first present one initial attempt at illustrating the difficulty of solving this issue via dataset resplitting.

Limitation of Model Selection. In this experiment, we see if an oracle model selection process

Target Eval Set	MNLI-m	BREAK	HANS	SNLI-hard	STR-L	STR-S	STR-NE	STR-O	STR-A	STR-NU	SICK	EQU-NAT	EQU-SYN
Accuracy Mean													
MNLI-m	85.1	95.3	61.6	80.9	**81.9**	77.3	55.5	59.9	62.9	41.1	57.3	60.1	41.3
Re-Split Dev	-	**96.2**	**64.3**	**81.0**	81.7	**77.4**	**56.5**	**66.0**	**67.2**	**48.2**	**59.3**	**61.2**	**47.6**
Accuracy Standard Deviation													
MNLI-m	0.22	0.37	1.57	**0.33**	0.36	**0.35**	**0.65**	**0.88**	**1.60**	3.49	**0.55**	**1.06**	3.19
Re-Split Dev	-	**0.32**	**1.51**	0.52	**0.34**	0.47	0.83	2.70	1.83	**2.64**	1.26	1.18	**1.86**

Table 6: The comparison of means and standard deviations of the accuracies when model selection are conducted based on different development set. 'MNLI-m' chooses the best checkpoint based on the MNLI-m validation set. 'Re-Split Dev' chooses the best checkpoint based on the corresponding re-splitted analysis-dev set.

can help reduce instability. Unlike the benchmark datasets, such as SNLI, MNLI, and SQuAD, analysis sets are often proposed as a single set without dev/test splits. In Sec. 4, we observe that models' performances on analysis sets have little correlation with model performance on standard validation sets, making the selection model routine useless for reducing performance instability on analysis sets. Therefore, we do oracle model selection by dividing the original analysis set into an 80% analysis-dev dataset and a 20% analysis-test dataset. Model selection is a procedure used to select the best model based on the high correlation between dev/test sets. Hence, the dev/test split here will naturally be expected to have the best performance.

In Table 6, we compare the results of BERT-B on the new analysis-test with model selection based on the results on either MNLI or the corresponding analysis-dev. While model selection on analysis-dev helps increase the mean performance on several datasets[6], especially on HANS, STR-O, and STR-NU, indicating the expected high correlation inside the analysis set, however, the variances of final results are not always reduced for different datasets. Hence, besides the performance instability caused by noisy model selection, different random seeds indeed lead to models with different performance on analysis datasets. This observation might indicate that performance instability is relatively independent of the mean performance and hints that current models may have intrinsic randomness brought by different random seeds which is unlikely to be removed through simple dataset/model fixes.

5.1 Implications of Result Instability

If the intrinsic randomness in the model prevents a quick fix, what does this instability issue imply? At first glance, one may view the instability as a problem caused by careless dataset design or deficiency in model architecture/training algorithms. While both parts are indeed imperfect, here we suggest it is more beneficial to view this instability as an inevitable consequence of the current datasets and models. On the data side, as these analysis datasets usually leverage specific rules or linguistic patterns to generate examples targeting specific linguistic phenomena and properties, they contain highly similar examples (examples shown in 4.3). Hence, the model's predictions of these examples will be inevitably highly-correlated. On the model side, as the current model is not good enough to stably capture these hard linguistic/logical properties through learning, they will exhibit instability over some examples, which is amplified by the high correlation between examples' predictions. These datasets can still serve as good evaluation tools as long as we are aware of the instability issue and report results with multiple runs. To better handle the instability, we also propose some long and short term solution suggestions below, based on variance reporting and analysis dataset diversification.

5.2 Short/Long Term Suggestions

Better Analysis Reporting (Short Term). Even if we cannot get a quick fix to remove the instability in the results, it is still important to keep making progress using currently available resources, and more importantly, to accurately evaluate this progress. Therefore, in the short run, we encourage researchers to report the decomposed variance (Idp Var and Cov) for a more accurate understanding of the models and datasets as in Sec 4.2, Table 3 and Table 4. The first number (independent variance, i.e., Idp Var) can be viewed as a metric regarding how stable the model makes one single prediction and this number can be compared across different models. Models with a lower score can be interpreted as being more stable for one single

[6]Although the new selection increase the performance mean, we suggest not use the results on analysis sets as benchmark scores but only as toolkits to probe model/architecture changes since analysis datasets are easy to overfit.

prediction. The values of Cov also help us better understand both the model and the datasets. A high Cov indicates that many examples look similar to the model, and the model may be exploiting some common artifacts in this group of examples. A lower Cov usually means that the dataset is diverse and is preferable for evaluation. By comparing models with both total variance and the Idp Var, we can have a better understanding of where the instability of the models comes from. A more stable model should aim to improve the total variance with more focus on Idp Var. If the target is to learn the targeted property of the dataset better, then more focus should be drawn towards the second term when analysing the results.

Model and Dataset Suggestions (Long Term). In the long run, we should be focusing on improving models (including better inductive biases, large-scale pre-training with tasks concerning structure/compositionality) so that they can get high accuracy stably. Dataset-wise, we encourage the construction of more diverse datasets (in terms of syntax and lexicon). From our previous results and analysis in Section 4, we can see that analysis datasets from natural real-life sources usually lead to lower covariance between predictions and show better stability. Manual verification for synthetic examples also helps reduce the instability of analysis datasets. While controlled synthetic datasets are more accurate and effective in evaluating certain linguistic phenomenon, the lack of diversity may increase the model's ability to guess the answer right and solve only that single pattern/property instead of mastering the systematic capability of those linguistic properties under different contexts (as reflected by the poor correlation between different analysis datasets). Therefore, a very valuable direction in constructing these datasets is to both maintain the specificity of the dataset while having a larger diversity.

6 Conclusions

Auxiliary analysis datasets are meant to be important resources for debugging and understanding models. However, large instability of current models on some of these analysis sets undermine such benefits and bring non-ignorable obstacles for future research. In this paper, we examine the issue of instability in detail, provide theoretical and empirical evidence discovering the high inter-example correlation that causes this issue. Finally, we give suggestions on future research directions and on better analysis variance reporting. We hope this paper will guide researchers on how to handle instability and inspire future work in this direction.

Acknowledgments

We thank the reviewers for their helpful comments. This work was supported by ONR Grant N00014-18-1-2871, DARPA YFA17-D17AP00022, and NSF-CAREER Award 1846185. The views contained in this article are those of the authors and not of the funding agency.

References

Michael Bloodgood and John Grothendieck. 2013. Analysis of stopping active learning based on stabilizing predictions. In *Proceedings of the Seventeenth Conference on Computational Natural Language Learning*, pages 10–19.

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*, pages 632–642.

Vicente Iván Sánchez Carmona, Jeff Mitchell, and Sebastian Riedel. 2018. Behavior analysis of nli models: Uncovering the influence of three factors on robustness. *NAACL*.

Qian Chen, Xiaodan Zhu, Zhen-Hua Ling, Si Wei, Hui Jiang, and Diana Inkpen. 2017. Enhanced lstm for natural language inference. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1657–1668.

Christopher Clark, Mark Yatskar, and Luke Zettlemoyer. 2019. Don't take the easy way out: Ensemble based methods for avoiding known dataset biases. 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)*, pages 4060–4073.

Zihang Dai, Zhilin Yang, Yiming Yang, Jaime G Carbonell, Quoc Le, and Ruslan Salakhutdinov. 2019. Transformer-xl: Attentive language models beyond a fixed-length context. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2978–2988.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for*

Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers), pages 4171–4186.

Jesse Dodge, Gabriel Ilharco, Roy Schwartz, Ali Farhadi, Hannaneh Hajishirzi, and Noah Smith. 2020. Fine-tuning pretrained language models: Weight initializations, data orders, and early stopping. *arXiv preprint arXiv:2002.06305*.

Matt Gardner, Joel Grus, Mark Neumann, Oyvind Tafjord, Pradeep Dasigi, Nelson F Liu, Matthew Peters, Michael Schmitz, and Luke Zettlemoyer. 2018. Allennlp: A deep semantic natural language processing platform. In *Proceedings of Workshop for NLP Open Source Software (NLP-OSS)*, pages 1–6.

Atticus Geiger, Ignacio Cases, Lauri Karttunen, and Chris Potts. 2019. Posing fair generalization tasks for natural language inference. *EMNLP*.

Mor Geva, Yoav Goldberg, and Jonathan Berant. 2019. Are we modeling the task or the annotator? an investigation of annotator bias in natural language understanding datasets. *EMNLP*.

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 (Volume 2: Short Papers)*, pages 650–655.

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 *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6904–6913.

Suchin Gururangan, Swabha Swayamdipta, Omer Levy, Roy Schwartz, Samuel Bowman, and Noah A Smith. 2018. Annotation artifacts in natural language inference data. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 107–112.

He He, Sheng Zha, and Haohan Wang. 2019. Unlearn dataset bias in natural language inference by fitting the residual. In *Proceedings of the 2nd Workshop on Deep Learning Approaches for Low-Resource NLP (DeepLo 2019)*, pages 132–142.

Alex Irpan. 2018. Deep reinforcement learning doesn't work yet. https://www.alexirpan.com/2018/02/14/rl-hard.html.

Allan Jabri, Armand Joulin, and Laurens Van Der Maaten. 2016. Revisiting visual question answering baselines. In *European conference on computer vision*, pages 727–739. Springer.

Robin Jia and Percy Liang. 2017. Adversarial examples for evaluating reading comprehension systems. *EMNLP*.

Divyansh Kaushik and Zachary C Lipton. 2018. How much reading does reading comprehension require? a critical investigation of popular benchmarks. *EMNLP*.

Mike Lewis and Angela Fan. 2019. Generative question answering: Learning to answer the whole question. In *ICLR*.

Nelson F Liu, Roy Schwartz, and Noah A Smith. 2019a. Inoculation by fine-tuning: A method for analyzing challenge datasets. *NAACL*.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019b. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Matthew C Makel, Jonathan A Plucker, and Boyd Hegarty. 2012. Replications in psychology research: How often do they really occur? *Perspectives on Psychological Science*, 7(6):537–542.

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*.

R Thomas McCoy, Junghyun Min, and Tal Linzen. 2019a. Berts of a feather do not generalize together: Large variability in generalization across models with similar test set performance. *arXiv preprint arXiv:1911.02969*.

Tom McCoy, Ellie Pavlick, and Tal Linzen. 2019b. Right for the wrong reasons: Diagnosing syntactic heuristics in natural language inference. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3428–3448.

Pasquale Minervini and Sebastian Riedel. 2018. Adversarially regularising neural nli models to integrate logical background knowledge. *CoNLL*.

Nafise Sadat Moosavi, Prasetya Ajie Utama, Andreas Rücklé, and Iryna Gurevych. 2019. Improving generalization by incorporating coverage in natural language inference. *arXiv preprint arXiv:1909.08940*.

Aakanksha Naik, Abhilasha Ravichander, Norman Sadeh, Carolyn Rose, and Graham Neubig. 2018a. Stress test evaluation for natural language inference. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 2340–2353.

Aakanksha Naik, Abhilasha Ravichander, Norman Sadeh, Carolyn Rose, and Graham Neubig. 2018b. Stress test evaluation for natural language inference. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 2340–2353, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Yixin Nie, Yicheng Wang, and Mohit Bansal. 2019. Analyzing compositionality-sensitivity of nli models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 6867–6874.

Yixin Nie, Adina Williams, Emily Dinan, Mohit Bansal, Jason Weston, and Douwe Kiela. 2020. Adversarial nli: A new benchmark for natural language understanding. *ACL*.

Deric Pang, Lucy H Lin, and Noah A Smith. 2019. Improving natural language inference with a pretrained parser. *arXiv preprint arXiv:1909.08217*.

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

Jason Phang, Thibault Févry, and Samuel R Bowman. 2018. Sentence encoders on stilts: Supplementary training on intermediate labeled-data tasks. *arXiv preprint arXiv:1811.01088*.

Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. Improving language understanding by generative pre-training. *URL https://s3-us-west-2. amazonaws. com/openai-assets/researchcovers/languageunsupervised/language understanding paper. pdf*.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners. *OpenAI Blog*, 1(8).

Pranav Rajpurkar, Robin Jia, and Percy Liang. 2018. Know what you don't know: Unanswerable questions for squad. *ACL*.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016a. SQuAD: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2383–2392, Austin, Texas. Association for Computational Linguistics.

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

Abhilasha Ravichander, Aakanksha Naik, Carolyn Rose, and Eduard Hovy. 2019. Equate: A benchmark evaluation framework for quantitative reasoning in natural language inference. In *Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL)*, pages 349–361.

Nils Reimers and Iryna Gurevych. 2017. Reporting score distributions makes a difference: Performance study of lstm-networks for sequence tagging. *EMNLP*.

Nils Reimers and Iryna Gurevych. 2018. Why comparing single performance scores does not allow to draw conclusions about machine learning approaches. *arXiv preprint arXiv:1803.09578*.

Ohad Rozen, Vered Shwartz, Roee Aharoni, and Ido Dagan. 2019. Diversify your datasets: Analyzing generalization via controlled variance in adversarial datasets. *CoNLL*.

Vered Shwartz and Ido Dagan. 2018. Paraphrase to explicate: Revealing implicit noun-compound relations. *ACL*.

Masatoshi Tsuchiya. 2018. Performance impact caused by hidden bias of training data for recognizing textual entailment. *LREC*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.

Alex Wang, Amanpreet Singh, Julian Michael, Felix Hill, Omer Levy, and Samuel R. Bowman. 2019. GLUE: A multi-task benchmark and analysis platform for natural language understanding. In *Proceedings of ICLR*.

Noah Weber, Leena Shekhar, and Niranjan Balasubramanian. 2018. The fine line between linguistic generalization and failure in seq2seq-attention models. *NAACL*.

Adina Williams, Nikita Nangia, and Samuel Bowman. 2018. A broad-coverage challenge corpus for sentence understanding through inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 1112–1122.

Yadollah Yaghoobzadeh, Remi Tachet, TJ Hazen, and Alessandro Sordoni. 2019. Robust natural language inference models with example forgetting. *arXiv preprint arXiv:1911.03861*.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Ruslan Salakhutdinov, and Quoc V Le. 2019. Xlnet: Generalized autoregressive pretraining for language understanding. *NeurIPS*.

Yi-Ting Yeh and Yun-Nung Chen. 2019. Qainfomax: Learning robust question answering system by mutual information maximization. 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)*, pages 3361–3366.

Appendix

A Details of Models

For models, we mainly focus on the current state-of-the-art models with a pre-trained transformer structure. In addition, we also selected several traditional models to see how different structures and the use of pre-trained representations influence the result.

A.1 Transformer Models

BERT (Devlin et al., 2019). BERT is a Transformer model pre-trained with masked language supervision on a large unlabeled corpus to obtain deep bi-directional representations (Vaswani et al., 2017). To conduct the task of NLI, the premise and the hypothesis are concatenated as the input and a simple classifier is added on top of these pre-trained representations to predict the label. Similarly, for RC, the question and the passage are concatenated as a single input and the start/end location of the answer span is predicted by computing a dot product between the start/end vector and all the words in the document. The whole model is fine-tuned on NLI/RC datasets before evaluation.

RoBERTa (Liu et al., 2019b). RoBERTa uses the same structure as BERT, but carefully tunes the hyper-parameters for pre-training and is trained 10 times more data during pre-training. The fine-tuning architecture and process are the same as BERT.

XLNet (Yang et al., 2019). XLNet also adopts the Transformer structure but the pre-training target is a generalized auto-regressive language modeling. It also can take in infinite-length input by using the Transformer-XL (Dai et al., 2019) architecture. The fine-tuning architecture and process are the same as BERT.

A.2 Traditional Models

ESIM (Chen et al., 2017). ESIM first uses BiLSTM to encode both the premise and the hypothesis sentence and perform cross-attention before making the prediction using a classifier. It is one representative model before the use of pre-trained Transformer structure.

B Details of Analysis Datasets

We used the following NLI analysis datasets in our experiments: **Break NLI** (Glockner et al., 2018), **SNLI-hard** (Gururangan et al., 2018), **NLI Stress Test** (Naik et al., 2018b) and **HANS** (McCoy et al., 2019b). We use **AdvSQuAD** (Jia and Liang, 2017) as the RC analysis dataset.

Name	Standard/Analysis	#Examples	#Classes
MNLI-m	Standard	9815	3
MNLI-mm	Standard	9832	3
SNLI	Standard	9842	3
BREAK-NLI	Analysis	8193	3
HANS	Analysis	30000	2
SNLI-hard	Analysis	3261	3
STR-L	Analysis	9815	3
STR-S	Analysis	8243	3
STR-NE	Analysis	9815	3
STR-O	Analysis	9815	3
STR-A	Analysis	1561	3
STR-NU	Analysis	7596	3
SICK	Analysis	9841	3
EQU-NAT	Analysis	1384	3
EQU-SYN	Analysis	8318	3

Table 7: Dataset statistics and categories for all the NLI dev/analysis datasets.

Name	Standard/Analysis	#Paragraphs	#Questions
SQuAD	Standard	48	10570
AddSent	Analysis	48	3560
AddOneSent	Analysis	48	1787

Table 8: Dataset statistics and categories for all the RC dev/analysis datasets.

Break NLI.[7] The examples in Break NLI resemble the examples in SNLI. The hypothesis is generated by swapping words in the premise so that lexical or world knowledge is required to make the correct prediction.

SNLI-Hard.[8] SNLI hard dataset is a subset of the test set of SNLI. The examples that can be predicted correctly by only looking at the annotation artifacts in the premise sentence are removed.

NLI Stress.[9] NLI Stress datasets is a collection of datasets modified from MNLI. Each dataset targets one specific linguistic phenomenon, including word overlap (STR-O), negation (STR-NE), antonyms (STR-A), numerical reasoning (STR-NU), length mismatch (STR-L), and spelling errors (STR-S). Models with certain weaknesses will get low performance on the corresponding dataset. In our experiments, we use the mismatched set if there

[7] github.com/BIU-NLP/Breaking_NLI

[8] nlp.stanford.edu/projects/snli/snli_1.0_test_hard.jsonl

[9] abhilasharavichander.github.io/NLI_StressTest/

Model	Standard Datasets			Analysis Sets											
	MNLI-m	MNLI-mm	SNLI	BREAK-NLI	HANS	SNLI-hard	STR-L	STR-S	STR-NE	STR-O	STR-A	STR-NU	SICK	EQU-NAT	EQU-SYN
ESIM	77.38±0.32	77.03±0.18	88.34±0.24	78.49±1.00	49.89±0.15	75.03±0.40	74.21±0.24	69.30±2.38	51.61±1.13	57.95±1.47	53.21±2.04	21.02±1.00	55.55±0.47	55.87±1.01	22.89±0.94
ESIM+ELMo	79.83±0.11	79.85±0.21	88.81±0.17	83.24±1.33	50.07±0.27	76.30±0.45	76.29±0.33	74.03±0.25	52.80±0.79	58.42±1.63	54.41±1.69	20.95±1.00	57.21±0.25	59.19±0.69	22.70±1.01
BERT-B	84.72 ±0.24	84.89 ±0.20	91.24 ±0.11	95.53±0.38	62.31±1.51	81.30±0.40	81.79±0.34	76.91±0.28	55.37±0.65	59.57±0.90	64.96±0.89	39.02±3.76	57.17±0.34	60.33±0.63	39.44±3.29
RoBERTa-B	87.64±0.12	87.66±0.17	91.94±0.07	97.04±0.36	72.45±1.02	82.44±0.30	85.13±0.15	81.97±0.27	57.39±0.63	63.38±0.98	73.84±1.61	52.80±3.39	57.14±0.32	63.92±0.67	51.85±2.71
XLNet-B	86.78±0.28	86.42±0.14	91.54±0.11	95.95±0.63	66.29±1.08	81.35±0.37	84.40±0.17	80.33±0.28	57.18±0.56	63.70±2.04	75.70±1.48	40.32±4.31	56.66±0.22	61.79±0.83	39.93±3.91
BERT-L	86.62±0.17	86.75±0.19	92.09±0.09	95.71±0.53	72.42±1.78	82.26±0.40	84.20±0.22	79.32±0.48	62.25±1.55	64.48±1.71	72.28±1.01	49.56±4.20	57.19±0.29	62.66±0.64	49.38±3.76
RoBERTa-L	90.04±0.17	89.99±0.15	93.09±0.12	97.50±0.19	75.90±0.99	84.42±0.30	87.68±0.19	85.67±0.22	60.03±2.04	63.10±1.71	78.96±1.91	61.27±5.25	57.77±0.29	66.11±0.55	58.34±4.13
XLNet-L	89.48±0.20	89.31±0.18	92.90±0.14	97.57±0.23	75.75±1.22	83.55±0.30	87.33±0.18	84.30±0.32	60.46±3.25	67.47±2.37	84.26±2.14	62.14±3.63	57.33±0.30	63.56±0.70	60.45±4.33

Table 9: Means and standard deviations of final performance on NLI datasets for all models.

Model	Standard Dataset	Analysis Sets	
	SQuAD	AddSent	AddOneSent
BERT-B	87.16±0.13	63.70±0.57	72.33±0.48
XLNet-B	89.33±0.39	69.19±1.18	77.20±0.94

Table 10: Means and standard deviations of final F1 on SQuAD dev set for both BERT-B and XLNet-B.

are both a matched version and a mismatched version. For STR-S, we follow the official evaluation script[10] to use the gram_content_word_swap subset.

HANS.[11] The examples in HANS are created to reveal three heuristics used by models: the lexical overlap heuristic, the sub-sequence heuristic, and the constituent heuristic. For each heuristic, examples are generated using 5 different templates.

SICK.[12] SICK is a dataset created for evaluating the compositional distributional semantic models. The sentences in this dataset come from the 8K ImageFlickr dataset and the SemEval 2012 STS MSR-Video Description dataset. The sentences are first normalized and then paired with an expanded version so that the pair can test certain lexical, syntactic, and semantic phenomena.

EQUATE.[13] EQUATE is a benchmark evaluation framework for evaluating quantitative reasoning in textual entailment. It consists of five test sets. Three of them are real-world examples (RTE-Quant, NewsNLI, RedditNLI) and two of them are controlled synthetic tests (AWPNLI, Stress Test). In this work, we use EQU-NAT to denote the real-world subset and EQU-SYN to denote the synthetic tests.

AdvSQuAD.[14] AdvSQuAD is a dataset created by inserting a distracting sentence into the original paragraph. This sentence is designed to be similar to the question but containing a wrong answer in order to fool the models.

C Dataset Statistics

Dataset statistics and categories for all the NLI datasets can be seen in Table 7. Dataset statistics and categories for all the RC datasets can be seen in Table 8.

D Training Details

For all pre-trained transformer models, namely, BERT, RoBERTa, and XLNet, we use the same set of hyper-parameters for analysis consideration. For NLI, we use the suggested hyper-parameters in Devlin et al. (2019). The batch size is set to 32 and the peak learning rate is set to 2e-5. We save checkpoints every 500 iterations, resulting in 117 intermediate checkpoints. In our preliminary experiments, we find that tuning these hyper-parameters will not significantly influence the results. The training set for NLI is the union of SNLI (Bowman et al., 2015) and MNLI (Williams et al., 2018)[15] training set and is fixed across all the experiments. This will give us a good estimation of state-of-the-art performance on NLI that is fairly comparable to other analysis studies. For RC, we use a batch size of 12 and set the peak learning rate to 3e-5. RC Models are trained on SQuAD1.1[16] (Rajpurkar et al., 2016b) for 2 epochs. All our experiments are run on Tesla V100 GPUs.

E Means and Standard Deviations of Final Results on NLI/RC datasets

Here we provide the mean and standard deviation of the final performance over 10 different seeds

[10] github.com/AbhilashaRavichander/NLI_StressTest/blob/master/eval.py

[11] github.com/tommccoy1/hans

[12] marcobaroni.org/composes/sick.html

[13] github.com/AbhilashaRavichander/EQUATE

[14] Both AddSent and AddOneSent can be downloaded from worksheets.codalab.org/worksheets/0xc86d3ebe69a3427d91f9aaa63f7d1e7d/.

[15] Both SNLI and MNLI can be downloaded from gluebenchmark.com.

[16] rajpurkar.github.io/SQuAD-explorer/

Original Context:	In February 2010, in response to controversies regarding claims in the Fourth Assessment Report, five climate scientists–all contributing or lead IPCC report authors–wrote in the journal Nature calling for changes to the IPCC. They suggested a range of new organizational options, from tightening the selection of lead authors and contributors to dumping it in favor of a small permanent body or even turning the whole climate science assessment process into a moderated "living" Wikipedia-IPCC. Other recommendations included that the panel employs full-time staff and remove government oversight from its processes to avoid political interference.
Question:	How was it suggested that the IPCC avoid political problems?
Answer:	remove government oversight from its processes
Distractor Sentence 1:	It was suggested that the PANEL avoid nonpolitical problems.
Distractor Sentence 2:	It was suggested that the panel could avoid nonpolitical problems by learning.

Table 11: A highly-correlated example pair in the SQuAD-AddSent dataset based with the BERT model. This example pair have the largest covariance (0.278) among all the pairs.

on NLI and RC datasets in Table 9 and Table 10 respectively.

F High-Correlated Cases for SQuAD

In this section, we show an example to illustrate that the high-correlated cases are similar to NLI datasets for RC datasets. As adversarial RC datasets such as AddSent are created by appending a distractor sentence at the end of the original passage, different examples can look very similar. In Table 11, we see two examples are created by appending two similar distractor sentences to the same context, making the predictions of these two examples highly correlated.