Adversarial Analysis of Natural Language Inference Systems

Tiffany Chien
University of California, Berkeley

Jugal Kalita
University of Colorado, Colorado Springs

Abstract

The release of large natural language inference (NLI) datasets like SNLI and MNLI have led to rapid development and improvement of completely neural systems for the task. Most recently, heavily pre-trained, Transformer-based models like BERT and MT-DNN have reached near-human performance on these datasets. However, these standard datasets have been shown to contain many annotation artifacts: features that correlate strongly with the correct label in training (and testing), but are clearly non-generalizable (e.g. sentence length, word overlap). This allows models to shortcut understanding using simple fallible heuristics, and still perform well on the test set. So it is no surprise that many adversarial (challenge) datasets have been created that cause models trained on standard datasets to fail dramatically. Although extra training on this data generally improves model performance on just that type of data, transferring that learning to unseen examples is still partial at best. This work evaluates the failures of state-of-the-art models on existing adversarial datasets that test different linguistic phenomena, and find that even though the models perform similarly on MNLI, they differ greatly in their robustness to these attacks. In particular, we find syntax-related attacks to be particularly effective across all models, so we provide a fine-grained analysis and comparison of model performance on those examples. We draw conclusions about the value of model size and multi-task learning (beyond comparing their standard test set performance), and provide suggestions for more effective training data.

Introduction

In recent years, deep learning models have achieved and continued to improve on state-of-the-art results on many NLP tasks. However, models that perform extremely well on standard datasets have been shown to be rather brittle and easily tricked. In particular, the idea of *adversarial* examples or attacks was brought over from computer vision, and various methods of slightly perturbing inputs have been developed that cause models to fail catastrophically (McCoy, Pavlick, and Linzen 2019; Glockner, Shwartz, and Goldberg 2018; Naik et al. 2018).

Adversarial attacks need to be studied from a security perspective for the deployment of real-world systems, but they are also a powerful lens into *interpretability* of black-box deep learning systems. By examining the failures of state-of-the-art models, we can learn a lot about what they are really learning, which may give us insights into improving their robustness and general performance.

One philosophical generalization about the cause of failure for all current NLP systems is a lack of deep, 'real' understanding of language. We will focus on the task of natural language inference (NLI), which is a basic natural language understanding task thought to be a key stepping stone to higher-level understanding tasks like question answering and summarization. The setup of the NLI task is to determine whether a *hypothesis* is true given a *premise*, answering *entailment*, *contradiction*, or *neutral*.

The current top-performing systems for NLI rely on pre-training on generic tasks, followed by fine-tuning on a labeled task-specific dataset. This is in contrast to older (before late 2018) models, which were primarily task-specific architectures trained primarily on task-specific labeled datasets. In addition, the Transformer architecture (Vaswani et al. 2017) now outperforms the previously dominating recurrent architectures (LSTM and variants). We want to analyze what kinds of adversarial attacks are still potent on highly-acclaimed recent NLP models like BERT (Devlin et al. 2018) and MT-DNN (Liu et al. 2019).

Our contributions are as follows:

- We test models on a variety of existing adversarial datasets, with a high level of granularity to different linguistic phenomena. Results indicate that the pre-trained models are remarkably good at lexical meaning, while struggling most with logic and syntactic phenomena.
- We focus in on the syntax-focused dataset created by McCoy, Pavlick, and Linzen. We look closely at the 30 subcases, and analyze the effects of model size (base vs. large size) and multi-task learning (MT-DNN vs. BERT). We also examine what subcases all models fail at.
- We experiment with fine-tuning the models with (flattened) dependency parses as input (with no adjustments to architecture or data pre-processing). We find that this does improves performance on some, but not all, subcases that rely on the hierarchical structure of sentences.
- Lastly, we investigate MNLI's biases by analyzing performance after different amounts of fine-tuning (more and more overfitting) on MNLI.

Related Work

This work joins a growing movement in NLP to go beyond improving test set metrics to more deeply analyze model learning and performance (Belinkov and Glass 2019). This genre of work believes in the value of interpretability, both to build safer practical systems, and just to find fruitful directions for improving raw model performance.

Liu, Schwartz, and Smith (2019) use a metaphor of inoculation to disentangle the blame for adversarial vulnerability between training data and model architecture. They expose a small part of the challenge dataset to the model during training, and re-test its evaluation performance on the original test set and the challenge dataset.

1. If the model still fails the challenge dataset, the weakness probably lies in its design/architecture or training process.

2. If the model can now succeed at the challenge dataset (without sacrificing performance on the original dataset), then the original dataset is at fault.

3. If the model does better on the challenge dataset but worse on the original dataset, the challenge dataset is somehow not representative of the phenomenon it was trying to test, for example having annotation artifacts or being very skewed to a particular label.

Unfortunately, even if adversarial training does improve model performance on that particular dataset, it is fundamentally impossible to devise and train on all possible linguistic phenomena. The transferability of adversarial robustness to new kinds of examples has been tested by some of the creators of adversarial datasets, by withholding some example generation methods while training on others. Nie, Wang, and Bansal (2018) find that knowledge of each of their rule-based templates was almost completely non-transferable to others. In fact, training on some specific templates caused overfitting and hurt overall robustness. McCoy, Pavlick, and Linzen (2019) find more mixed results, with some cases of successful transfer.

Many standard datasets for different tasks have been shown to have blatant annotation artifacts, allowing models to learn features that are strong in the training (and testing) data, but that have nothing to do with actually performing the task. Gururangan et al. (2018) find many of these artifacts in standard NLI datasets (SNLI and MNLI). For example, *neutral* hypotheses tend to be longer in length, because an easy way to generate a hypothesis that isn't necessarily entailed by the premise is to add extra details. Meanwhile, strong negation words like *nobody, no, never* are strong indicators of *contradiction*. With these artifacts in mind, they split the data into "hard" and "easy" versions, and model performance decreased by about 15% on the hard test set. These findings suggest that it is not the models' faults for failing on adversarial examples, given that there exist easier ways to get high accuracy than truly understanding anything. But it also means that current evaluation metrics greatly overestimate models' abilities and understanding.

Models

The two new models that we study gain most of their power from pre-training on a generic language task with a huge unlabeled dataset. They achieve state-of-the-art performance on a variety of language understanding tasks.

1. **BERT** (Devlin et al. 2018) pre-trains on a bidirectional word-masking language modelling task, in addition to sentence pair prediction, i.e. whether the second sentence is likely to directly follow the first.

2. **MT-DNN** (Liu et al. 2019) builds on BERT by performing multi-task learning on the nine GLUE (General Language Understanding Evaluation) benchmark tasks (Wang et al. 2018), after BERT's pre-training.

BERT is based on the Transformer architecture (Vaswani et al. 2017), a non-recurrent, purely attention-based architecture. BERT has a base version (12 Transformer layers), and a large version (24 layers). We trained base and large versions of both BERT and MT-DNN. These models are fine-tuned on MNLI starting from publicly available pre-trained checkpoints.

We compare with an older recurrent model, ESIM (Enhanced Sequential Inference Model) (Chen et al. 2016). It is NLI-task-specific and only trained on MNLI, with no huge pre-training. It uses a bidirectional LSTM to encode the premise and hypothesis sentences, and uses attention across those representations.

We also considered another model, Syntactic TreeLSTM (**S-TLSTM**), which is identical to ESIM except it uses a TreeLSTM that takes a dependency parse as input (Chen et al. 2016). This model may provide a useful comparison to BERT because its explicit use of the hierarchical structure of language is the exact opposite model design direction from extensive unsupervised pre-training. However, various studies suggest that the BERT architecture does in fact learn hierarchical structure: Goldberg (2019) found that BERT performed remarkably well when fine-tuned for external syntactic classification tasks, and Jawahar, Sagot, and Seddah (2019) showed that different layers of BERT learned structural representations of language at different abstraction levels. McCoy, Pavlick, and Linzen (2019) test a different tree-based model (SPINN (Bowman et al. 2016)) on their adversarial dataset, and find that it outperforms ESIM, but not BERT. Considering all this, and the fact that there is currently no tree-based model that comes close to outperforming BERT and variants on standard datasets, we decided not to test S-TLSTM, despite its philosophical appeal.

Overall Results and Analysis

First, for reference, we provide the accuracies on the matched MNLI dev set for the models we trained (and tested) in Table 1. BERT-large results do not quite match published results, but we had limited hardware and did not carefully tune hyperparameters. The BERT-based models all perform comparably, and even ESIM does respectably.

Let us now analyze the performance of the selected models on the adversarial datasets (also called challenge sets, stress tests). We discuss the first two briefly and then focus

Model	Accuracy (%)
ESIM	76.80
BERT base	84.17
BERT large	**85.84**
MT-DNN base	84.20
MT-DNN large	**86.69**

Table 1: Overall MNLI Results

on the last one (McCoy, Pavlick, and Linzen 2019) because it is the most interesting in terms of actually distinguishing the strengths of the better-performing models.

Glockner, Shwartz, and Goldberg (2018) This dataset is created by modifying SNLI examples with single word replacements of different lexical relations, based on WordNet. It tests lexical inferences and relatively simple world knowledge. They test a model called KIM (Knowledge-based Inference Model) (Chen et al. 2016), which builds on ESIM to explicitly incorporate knowledge from WordNet in a variety of ways, including in architecture additions. However, the BERT-based models still significantly outperform KIM. This could be due to model architecture, but is most likely a result of their extensive pretraining on a huge diverse corpus. There is not a big difference between model sizes, or between MT-DNN and BERT. This suggests that lexical semantics is more basic and low-level, so learning it does not need so many layers of abstraction, or multi-task learning (see Table 2).

Model	Accuracy (%)
ESIM*	65.6
KIM*	83.5
BERT base	92.2
BERT large	**94.2**
MT-DNN base	92.9
MT-DNN large	**94.8**

Table 2: Single Word Replacement Attacks from (Glockner, Shwartz, and Goldberg 2018). ESIM and KIM results from original paper.

Naik et al. (2018) This dataset is composed of a variety of tests motivated by a manual examination and categorization of 100 mistakes made by the best performing model at the time (Nie and Bansal 2017). The categories are antonyms, word overlap (append "and true is true"), negation words (append "and false is not true"), length mismatch (append "and true is true" 5 times), and spelling errors. Antonyms and Spelling are "competence" tests, while the rest are "distraction" tests. The examples are generated by modifying examples from MNLI. We report accuracy averaged over all categories in Table .

$\text{BERT}_{\text{large}}$ and $\text{MT-DNN}_{\text{large}}$ do best. Overall model performance trends the same as performance on MNLI, but

Model	Accuracy (%)
ESIM	68.39
BERT base	74.30
BERT large	**77.21**
MT-DNN base	73.73
MT-DNN large	**77.14**

differences are not huge. Furthermore, when we examined performance on specific categories, all models had about the same pattern of relative performance on different categories of tests, i.e. they have the same relative successes and failures. This consistency and generally similar performance indicates in this case that the dataset is not well-targeted enough for really interesting insight. In addition, compared to McCoy, Pavlick, and Linzen (2019) (below), the way that examples are generated is more artificial, and maybe less meaningful. Of course, a robust NLI system still should not be defeated by this kind of attack, i.e. be able to determine irrelevant information, including tautologies, and this test shows that even the best models do not have this capability mastered properly.

McCoy, Pavlick, and Linzen (2019) They hypothesize that models utilize shallow, fallible syntactic heuristics to achieve accuracy on MNLI, instead of "real" understanding. The dataset consists of examples generated from manually created templates that break these heuristics. They have three categories of heuristics (each is a special case of the one before).

1. Lexical overlap: Model is likely to answer *entailment* if the premise and hypothesis share a lot of words.
   Would trick bag-of-words (no word order) models.
2. Subsequence: The hypothesis is a contiguous string of words from the premise.
   The ball by the bed rolled. $\nrightarrow$ *The bed rolled.*
   Could confuse sequence models too.
3. Constituent: The hypothesis is a syntactic constituent in the premise.
   If the boys slept, they would not eat. $\nrightarrow$ *The boys slept.*
   Could confuse models that know about syntax.

All three heuristics involve the model thinking the answer is *entailment* when it is not, i.e. the *non-entailment* examples are the ones that contradict the heuristic. So the extreme imbalance in model performance between entailment and non-entailment examples is strong evidence that the models do indeed rely on the hypothesized heuristics (Table 3 vs. 4).

Entailment	word overlap	subseq	constituent
ESIM	96.52	98.46	94.48
$\text{BERT}_{\text{base}}$	97.20	99.52	99.04
$\text{BERT}_{\text{large}}$	**90.48**	99.48	96.70
$\text{MT-DNN}_{\text{base}}$	97.22	99.98	99.22
$\text{MT-DNN}_{\text{large}}$	96.06	99.54	99.14

Table 3: Accuracy on examples labeled 'entailment'

Non-entailment	word overlap	subseq	constituent
ESIM	1.56	4.88	3.32
$\text{BERT}_{\text{base}}$	54.68	9.46	4.88
$\text{BERT}_{\text{large}}$	**83.44**	**31.38**	**44.72**
$\text{MT-DNN}_{\text{base}}$	72.96	5.66	16.50
$\text{MT-DNN}_{\text{large}}$	**88.08**	**31.24**	22.88

Table 4: Accuracy on examples labeled 'non-entailment'

All the BERT-based models do significantly better than the LSTM-based ESIM in most categories, as we see in Table 4. But $\text{BERT}_{\text{large}}$ and $\text{MT-DNN}_{\text{large}}$ do vastly better than all others, a difference that was not nearly as apparent in any of the other datasets we tested. In combination with the granularity in the manually created templates, these huge differences in performance indicate that this dataset more directly probes and reveals the strengths and weaknesses of different models.

The success of $\text{BERT}_{\text{large}}$ and $\text{MT-DNN}_{\text{large}}$ suggests that structural/syntactic information is learned more deeply by a larger model with more layers and parameters to work with (in contrast to lexical semantics (Glockner, Shwartz, and Goldberg, above)). $\text{BERT}_{\text{large}}$ also has lower accuracy on the *entailment* examples, also indicating that it is less prone to blindly following the heuristics.

$\text{MT-DNN}_{\text{base}}$ (which is built on $\text{BERT}_{\text{base}}$ and is therefore of comparable size) does significantly better than $\text{BERT}_{\text{base}}$ in some categories, indicating the value of multi-task learning (specifically on language understanding tasks).

Fine-grained Model Comparison

Comparison of $\text{BERT}_{\text{base}}$ and $\text{BERT}_{\text{large}}$

$\text{BERT}_{\text{large}}$ performs better than or equal to $\text{BERT}_{\text{base}}$ (at worst -1%) on all fifteen *non-entailment* subcases. Some templates saw particularly large improvement, such as modifying clauses:

- Relative clauses that modify nouns (+42.4%)
  The artists that supported the senators shouted. ↛ The senators shouted.
- Prepositional phrase modifiers (+38%)
  The managers next to the professors performed. ↛ The professors performed.

Understanding modifying clauses requires understanding the mechanics of compositional semantics (probably utilizing some kind of hierarchical syntax), which is a basic but crucial step in language understanding. So BERT-large's performance over BERT-base on these examples is evidence of significantly deeper understanding.

Another area of improvement is the lexical meanings of special subclasses of verbs and adverbs.

- Non-truth verbs with clause complements (+60.4%)
  The tourists said that the lawyer saw the secretary. ↛ The lawyer saw the secretary.
  This template uses a variety of verbs, all of which suggest but do not entail their complements.
- Modal adverbs (+26.7%)
  Maybe the scientist admired the lawyers. ↛ The scientist admired the lawyers.

Similarly, passive voice is a special *syntactic* phenomenon that BERT-large improves on, but still has trouble with.

- Passive voice (3.6% → 29.8%)
  The managers were advised by the athlete. ↛ The managers advised the athlete.

$\text{BERT}_{\text{base}}$ and $\text{BERT}_{\text{large}}$ were trained (pre-training and fine-tuning) on the same data, so the difference in the richness of their learning must reside only in the doubled number of layers in $\text{BERT}_{\text{large}}$. These performance improvements are evidence that more layers is necessary space for learning all the different special cases of language.

There are also some partially learned special cases, such as the meaning of "if" and related (logical implication).

- 76.6% → 98.7%: *Unless the professor danced, the student waited. ↛ The professor danced.*
- both 0%: *Unless the bankers called the professor, the lawyers shouted. ↛ The lawyers shouted.*

Meanwhile, all models fail to understand the logical meaning of disjunction (0-2%).

- *The actor helped the lawyers, or the managers stopped the author. ↛ The actor helped the lawyers.*

Logic is a very important component of inference as an understanding task, but understandably difficult for statistical models to learn properly, because it is in some sense not probabilistic, in addition to being dependent on exact meanings of single function words. Many traditional inference systems relied primarily on formal logic machinery, and finding a way to incorporate that into new models seems like a promising direction. Designing and training neural networks that parse and understand formal, symbolic logic is a pretty well-studied problem (Evans et al. 2018), and it is certainly known theoretically that general neural networks can represent arbitrary nonlinear logical relations. The difficulty is getting natural language models to actually care enough about logic during training to use it correctly for a specific task. Many different approaches have been explored recently, including but not limited to modifying the loss function to encourage logical consistency (Minervini and Riedel 2018), rule distillation in a teacher-student network (Hu et al. 2016), and indirect supervision using probabilitic logic (Wang and Poon 2018). To our knowledge, these have not yet been incorporated into state-of-the-art models, but they show promising results on the baseline models tested, especially in lower-resource scenarios.

All of these special cases are almost certainly encountered in BERT's huge pre-training corpus, but that unsupervised stage does not necessarily teach the model how to use that information towards performing inference. This is why larger and larger pre-training may not be the most effective or at least efficient way to achieve language understanding.

Some of the subsequence templates are still a struggle for all models, including large BERT and MT-DNN (<10%):

- *The manager knew the athlete mentioned the actor ↛ The manager knew the athlete.*

Heuristic	Syntactic subcategory	MT-DNN large	BERT large	MT-DNN base	BERT base	ESIM	BERT large UP	MT-DNN base PO
Lexical Overlap	subject/object_swap	**0.999**	**0.994**	0.935	0.729	0	0.988	0.936
	preposition	0.934	**0.979**	0.794	0.745	0.004	0.960	0.889
	relative_clause	0.912	**0.928**	0.699	0.504	0.069	**0.930**	0.837
	passive	**0.625**	0.298	0.432	0.036	0	0.214	0.505
	conjunction	0.934	**0.973**	0.788	0.720	0.005	0.943	0.711
Subseq	NP/S	0.042	0.003	0	0.016	0.058	0.004	0.003
	PP_on_subject	0.668	0.673	0.168	0.293	0.001	**0.786**	0.533
	relative_clause_on_subject	0.749	0.698	0.082	0.133	0.087	**0.863**	0.347
	past_participle	0.006	0.049	0.013	0.018	0.050	0.032	0.008
	NP/Z	0.097	0.146	0.020	0.013	0.047	**0.217**	0.172
Constituent	embedded_under_if	0.703	**0.987**	0.369	0.767	0.137	0.907	0.387
	after_if_clause	0.001	0	0	0	0	0	0.010
	embedded_under_verb	0.342	**0.903**	0.252	0.299	0	0.546	0.146
	disjunction	0.005	0	0.001	0.001	0.029	0.008	0.002
	adverb	0.093	**0.346**	0.203	0.079	0	0.083	0.036

Table 5: Results for *non-entailment* subcases. Each row corresponds to a syntactic phenomenon. BERT large UP: trained on unparsed then parsed; MT DNN-base PO: trained on parsed only

- *When the students fought the secretary ran. ↛ The students fought the secretary.*

These templates are in the spirit of *garden path sentences*, where local syntactic ambiguity causes a sequential reading of a sentence to lead to an incorrect interpretation. This kind of sentence has been studied extensively in cognitive science, specifically language processing, as human readers are first misled and then must backtrack to reanalyze the composition of the sentence to understand it properly (Ferreira and Henderson 1991; Osterhout, Holcomb, and Swinney 1994). Goldberg (2019) shows that BERT performs well on complex subject-verb agreement tasks, even without any fine-tuning, indicating that the pre-trained model already has the ability to correctly parse this kind of sentence. So the model somehow knows about syntax but does not know how to use it towards the task of inference, a teaching failure that can only be blamed on the inference-task-specific fine-tuning. MNLI probably has a low occurrence of complex syntax, but perhaps more importantly, the complete syntactic information is rarely necessary to perform the task. Nevertheless, an ability to utilize challenging syntax is an important generalizable skill, because it indicates deep, principled understanding of language.

Comparison of BERT and MT-DNN

Even though MT-DNN$_{large}$ performs better on MNLI than BERT$_{\text{large}}$, BERT beats MT-DNN on more subcases in this dataset. In particular, MT-DNN$_{large}$ struggles much more with subcases that test special lexical meanings that prevent entailment (number is difference between MT-DNN$_{large}$ and BERT$_{\text{large}}$):

1. conditionals: if, unless, whether or not (28.4%)
2. 'belief' verbs: believed, thought, hoped (56.1%)
3. uncertainty adverbs: hopefully, maybe, probably (25.3%)

The only subcase that MT-DNN$_{large}$ is significantly better at is the passive voice (+32.7%).

MT-DNN is trained starting with a pre-trained BERT and then fine-tuning on the 9 language understanding tasks in the GLUE benchmark (before fine-tuning again on MNLI). So if MT-DNN performs worse than a BERT model of the same size, this fine-tuning caused it to *forget* some knowledge that it had before. This would happen if the datasets being fine-tuned on do not explicitly test that knowledge, teaching the model to care less about the information from these words. Considering that most of the GLUE tasks are not straight NLI tasks, it is somewhat unsurprising that the model forgot how these words affect entailment.

Parses as Input

Considering that syntactic phenomena are one of the models' weaknesses, we conduct an experiment of simply passing the flattened binary parses as the input "sentences". We use the automatically generated parses that come with MNLI and the adversarial dataset. We test on the dataset from McCoy, Pavlick, and Linzen (2019).
We try two fine-tuning regimens:

1. Fine tune on original (unparsed) MNLI, then fine-tune again on the same data, parsed (labeled UP in Table 5).
2. Only fine-tune on parsed MNLI (no other inference-specific fine-tuning) (labeled PO in Table 5).

We find that it is rather difficult to get the different models to train well. Some had loss that never converged, some got near 0% on all *non-entailment* subcases. The only reasonable parsed models are BERT$_{\text{large}}$ under the first regimen (UP), and MT-DNN$_{\text{base}}$ under the second (PO). It is likely that these difficulties could be overcome with some systematic hyperparameter tuning, but we see substantial consistency (in model performance on the adversarial dataset) between the two successes, so do not think it would be very in-

Type	Sentence 1	Sentence 2
NP/S	The manager knew the tourists supported the author.	The manager knew the tourists.
NP/Z	Since the judge stopped the author contacted the managers.	The judge stopped the author.
past_participle	The scientist presented in the school stopped the artists.	The scientist presented in the school.
after_if_clause	Unless the scientists introduced the presidents, the athletes recommended the senator.	The athletes recommended the senator.

Table 6: Non-entailed cases where $\mathrm{BERT}_{\mathrm{large}}$ does very poorly: Sentence 1 does not entail Sentence 2.

sightful to test more. But the fact that the models responded so differently to fine-tuning suggests that the models have significantly different 'knowledge states' in terms of what they learned about how to solve tasks, i.e. they ended up in different local optima after pre-training. This idea deserves more analysis, because the whole point of huge pre-training is to learn maximally transferable and general representations of language. Thus, how to guide models towards these ideal local optima (and away from overfitting) is a very important and difficult question.

The fact that any model is able to learn what to do with parses is already surprising, given that none of their pre-training is parsed. Evaluating on the parses of MNLI (matched dev set), $\mathrm{BERT}_{\mathrm{large}}$ achieves 82% accuracy (compare to 86% unparsed), and $\mathrm{MT\text{-}DNN}_{\mathrm{base}}$ gets 84% (equal to unparsed).

These are the six subcases that saw a 10% or greater change in accuracy between parsed and unparsed inputs. Numbers are percent change from unparsed to parsed ($\mathrm{BERT}_{\mathrm{large}}$, $\mathrm{MT\text{-}DNN}_{\mathrm{base}}$).
Parsing does better on:

- Modifiers on subject
  The managers next to the professors performed. $\nrightarrow$ *The professors performed.* (+11.3, +36.5)
  The artists that supported the senators shouted. $\nrightarrow$ *The senators shouted.* (+16.5, +26.5)
- NP/Z (+7.1, +15.2)
  Since the athlete hid the secretaries introduced the president. $\nrightarrow$ *The athlete hid the secretaries.*
  The parsed models still only achieve 21.7% and 17.2% accuracy, but this is still some improvement.
- Conjunction (+22.2, +1.8 (unparsed $\mathrm{MT\text{-}DNN}_{\mathrm{base}}$ already gets 90.8))
  The tourists and senators admired the athletes $\rightarrow$ *The tourists admired the athletes.*
  This is an *entailment* template, so $\mathrm{BERT}_{\mathrm{large}}$'s lower accuracy actually indicates less heuristic reliance, and parsed improvement from 64.4 $\rightarrow$ 86.6 really indicates better understanding (while $\mathrm{MT\text{-}DNN}_{\mathrm{base}}$'s performance could just be using the heuristic).

Parsing does worse on:

- Embedded clause under non-truth verb (-35.7, -10.6)
  The lawyers believed that the tourists shouted. $\nrightarrow$ *The tourists shouted.*
- Adverbs indicating uncertainty (-26.3, -16.7)
  Hopefully the presidents introduced the doctors $\nrightarrow$ *The presidents introduced the doctors.*

Of this small set of significant changes, it can be said that the parsed inputs helped the model with syntactic, hierarchical examples, and hurt it on specific lexical semantics. This is a surprisingly intuitive result: the model shifted its focus more to syntax!

However, these are the only subcases that changed significantly, out of 30, suggesting either that the parses don't encode that much useful information, or (more likely) that the fine-tuning didn't teach the model how to use the extra information. For example, maybe $\mathrm{BERT}_{\mathrm{large}}$ (trained on unparsed then the exact same data parsed) just learned to ignore parentheses.

Furthermore, the subcases which had score close to 0 for the unparsed model basically did not see any improvement. These obstinate cases are given in Table 6. Most of these cases are tests of syntactic phenomena, so parsed data certainly contains useful information, but again, the fine-tuning is somehow not enough to teach the model how to use it.

We do not think that parsing is necessarily a preprocessing step that should be incorporated into future models/systems, because it takes extra computational and annotated data resources. But this experiment does show that without induced biases, BERT's massive, generic pre-training does not capture some basic rule-like principles.

Overfitting to MNLI

Models learn and use fallible heuristics only because it works on their training datasets; in other words, they are overfit to their training data, MNLI. We analyze this process by evaluating the model after different amounts of fine-tuning on MNLI. We perform this experiment on $\mathrm{MT\text{-}DNN}_{large}$, the best performer on MNLI, and gauge overfitting by evaluating on the adversarial dataset from McCoy, Pavlick, and Linzen (non-entailment subcases).

Epoch #	1	2	3
matched MNLI dev set	85.66	86.69	86.59
McCoy *non-entailment*	44.09	47.40	42.49

Table 7: Accuracy (%) for $\mathrm{MT\text{-}DNN}_{large}$ fine-tuned on MNLI for varying numbers of epochs, and then evaluated on the dataset from McCoy, Pavlick, and Linzen.

The model trains very quickly, reaching 1% away from max dev accuracy after only one epoch of fine-tuning, and decreasing slightly on dev accuracy by the third epoch. This is a claimed benefit of multi-task learning: the model is more flexible to learning different tasks quickly.

From epoch 2 to 3, MNLI dev performance decreases by only 0.1%, but according to performance on the adversarial dataset, the model is relying significantly more on heuristics, revealing a more overfit state. Looking at specific subcases, the epoch-3 model differs by more than 10% in 6 subcases, split very similarly to what happened with parsed inputs:

- Improves at lexical semantics: 'belief' verbs (believed, thought) (+11.8%) and uncertainty adverbs (hopefully, maybe) (+24.3%)
- Gets worse at structural/syntactic phenomena: passive voice (-24.4%), conjunction (-12.4%), and subject modifiers (PP (-15.6%), relative clauses (-19.1%))

Interestingly, the subcases that more MNLI fine-tuning helps are exactly the same as the ones that $\mathrm{BERT}_{\mathrm{large}}$ beats $\mathrm{MT\text{-}DNN}_{large}$ on. This strongly suggests that the purpose of these words is emphasized in MNLI: MT-DNN forgets about it while fine-tuning on other GLUE tasks, and more fine-tuning on MNLI makes it re-learn it.

On the other hand, the subcases that more fine-tuning hurts are all structural/syntax-focused, indicating that MNLI is biased against actually utilizing complex syntactic phenomena in a way that affects entailment (supporting the *syntactic* heuristic hypothesis of McCoy, Pavlick, and Linzen).

Creating feasibly-sized training datasets with "no biases" is impossible. Here we find some subtle examples in MNLI, emphasizing the sensitivity of these models to pick up on any useful signal. NLI is a very broad task, making it hard to define what a natural or representative input distribution would be, so dataset should depend on desired abilities and applications.

Conclusion

In this work, we use adversarial and challenge datasets to probe and analyze the failures of current state-of-the-art natural language inference models, comparing BERT and MT-DNN models of different sizes. Evaluating on these datasets distinguishes the actual understanding capabilities of the different models better than simply their performance on MNLI (the large dataset they were trained on). Our analysis is very fine-grained, targeting many specific linguistic phenomena. We find various improvements from larger model size and multi-task learning. We find that the most difficult examples for the best models are logic or syntax-based, including propositional logic and garden-path sentences. We experiment with passing parses as input to the out-of-the-box pre-trained models, and find that it does provide some improvement in examples that require understanding syntax, demonstrating the value of syntactic induced biases. We analyze what overfitting to MNLI looks like, and reveal some biases/artifacts in the dataset.

Some may argue that testing NLI systems on artificially challenging datasets is unfair and not useful, because it is not representative of their performance on naturalistic, real-world data. But even if the data humans naturally produce is not so difficult (because humans also are lazy and use heuristics), the difference is that we always *can* parse sentences correctly, utilizing rules and principles. And we intuitively know that ability is crucial to robust, trustworthy, and *real* language understanding.

Acknowledgement

The work reported in this paper is supported by the National Science Foundation under Grant No. 1659788. Any opinions, findings and conclusions or recommendations expressed in this work are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

References

Belinkov, Y., and Glass, J. 2019. Analysis Methods in Neural Language Processing: A Survey. *Transactions of the Association for Computational Linguistics* 7:49–72.

Bowman, S. R.; Gauthier, J.; Rastogi, A.; Gupta, R.; Manning, C. D.; and Potts, C. 2016. A Fast Unified Model for Parsing and Sentence Understanding. *arXiv:1603.06021 [cs]*. arXiv: 1603.06021.

Chen, Q.; Zhu, X.; Ling, Z.; Wei, S.; Jiang, H.; and Inkpen, D. 2016. Enhanced LSTM for Natural Language Inference. *arXiv:1609.06038 [cs]*. arXiv: 1609.06038.

Devlin, J.; Chang, M.-W.; Lee, K.; and Toutanova, K. 2018. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. *arXiv:1810.04805 [cs]*. arXiv: 1810.04805.

Evans, R.; Saxton, D.; Amos, D.; Kohli, P.; and Grefenstette, E. 2018. Can Neural Networks Understand Logical Entailment? *arXiv:1802.08535 [cs]*. arXiv: 1802.08535.

Ferreira, F., and Henderson, J. M. 1991. Recovery from misanalyses of garden-path sentences. *Journal of Memory and Language* 30(6):725–745.

Glockner, M.; Shwartz, V.; and Goldberg, Y. 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)*, 650–655. Melbourne, Australia: Association for Computational Linguistics.

Goldberg, Y. 2019. Assessing BERT's Syntactic Abilities. *arXiv:1901.05287 [cs]*. arXiv: 1901.05287.

Gururangan, S.; Swayamdipta, S.; Levy, O.; Schwartz, R.; Bowman, S. R.; and Smith, N. A. 2018. Annotation Artifacts in Natural Language Inference Data. *arXiv:1803.02324 [cs]*. arXiv: 1803.02324.

Hu, Z.; Ma, X.; Liu, Z.; Hovy, E.; and Xing, E. 2016. Harnessing Deep Neural Networks with Logic Rules. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2410–2420. Berlin, Germany: Association for Computational Linguistics.

Jawahar, G.; Sagot, B.; and Seddah, D. 2019. What Does BERT Learn about the Structure of Language? In *Proceedings of the 57th Conference of the Association for Computational Linguistics*, 3651–3657. Florence, Italy: Association for Computational Linguistics.

Liu, X.; He, P.; Chen, W.; and Gao, J. 2019. Multi-Task Deep Neural Networks for Natural Language Understanding. *arXiv:1901.11504 [cs]*. arXiv: 1901.11504.

Liu, N. F.; Schwartz, R.; and Smith, N. A. 2019. Inoculation by Fine-Tuning: A Method for Analyzing Challenge Datasets. *arXiv:1904.02668 [cs]*. arXiv: 1904.02668.

McCoy, T.; Pavlick, E.; and Linzen, T. 2019. Right for the Wrong Reasons: Diagnosing Syntactic Heuristics in Natural Language Inference. In *Proceedings of the 57th Conference of the Association for Computational Linguistics*, 3428–3448. Florence, Italy: Association for Computational Linguistics.

Minervini, P., and Riedel, S. 2018. Adversarially Regularising Neural NLI Models to Integrate Logical Background Knowledge. *arXiv:1808.08609 [cs, stat]*. arXiv: 1808.08609.

Naik, A.; Ravichander, A.; Sadeh, N.; Rose, C.; and Neubig, G. 2018. Stress Test Evaluation for Natural Language Inference. *arXiv:1806.00692 [cs]*. arXiv: 1806.00692.

Nie, Y., and Bansal, M. 2017. Shortcut-Stacked Sentence Encoders for Multi-Domain Inference. In *Proceedings of the 2nd Workshop on Evaluating Vector Space Representations for NLP*, 41–45. Copenhagen, Denmark: Association for Computational Linguistics.

Nie, Y.; Wang, Y.; and Bansal, M. 2018. Analyzing Compositionality-Sensitivity of NLI Models. *arXiv:1811.07033 [cs]*. arXiv: 1811.07033.

Osterhout, L.; Holcomb, P. J.; and Swinney, D. A. 1994. Brain potentials elicited by garden-path sentences: Evidence of the application of verb information during parsing. *Journal of Experimental Psychology: Learning, Memory, and Cognition* 20(4):786–803.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, L.; and Polosukhin, I. 2017. Attention Is All You Need. *arXiv:1706.03762 [cs]*. arXiv: 1706.03762.

Wang, H., and Poon, H. 2018. Deep Probabilistic Logic: A Unifying Framework for Indirect Supervision. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, 1891–1902. Brussels, Belgium: Association for Computational Linguistics.

Wang, A.; Singh, A.; Michael, J.; Hill, F.; Levy, O.; and Bowman, S. R. 2018. GLUE: A Multi-Task Benchmark and Analysis Platform for Natural Language Understanding. *arXiv:1804.07461 [cs]*. arXiv: 1804.07461.