Adversarial Attacks and Training on ELECTRA-small

Devon DeBalsi
The University of Texas at Austin - CS388 Natural Language Processing Fall 2023
devondebalsi@utexas.edu

Abstract

This project's goal is to improve the performance of ELECTRA-small (Clark et al., 2020), applied for natural language inference, on sentence pairs with significant lexical overlap. I show ELECTRA-small can adopt shallow lexical overlap-based heuristics when trained on an unmodified version of the Stanford Natural Language Inference (SNLI) corpus (Bowman et al., 2015). I propose two methods to improve performance of ELECTRA-small trained on the SNLI corpus on sentence pairs with lexical overlap: dynamically generating adversarial examples during training, and statically constructing a training corpus that combines SNLI and sentence pairs with lexical overlap. Using a development dataset containing sentence pairs with lexical overlap (McCoy et al., 2019), the former method led to no significant improvement on accuracy compared to training on unmodified SNLI, staying at roughly 50% on a dataset from McCoy et al. (2019). The latter method boosted accuracy to nearly 100% on the same data from McCoy et al. (2019).

1 Introduction

Transformer-based models can capture alignment between words, because self-attention mechanisms allow all words in a phrase to attend to each-other (Vaswani et al., 2017). Therefore, transformer-based models are a common choice for natural language inference (NLI) (MacCartney and Manning, 2008) tasks. The NLI task is defined as follows: given a premise sentence, identify the most likely of three labels *entailment*, *contradiction*, and *neutral* for a hypothesis sentence; put simply, assigning probability to if a hypothesis is true given a premise. Literature (Zhou and Bansal, 2020), (Mendelson and Belinkov, 2021), (McCoy et al., 2019) shows that despite the self-attention mechanism, transformer-based models are still subject to adopting heuristics that subvert the expectation of capturing alignment between words – specifically, heuristics based on lexical overlap between sentence pairs.

The baseline model used for NLI for this project, ELECTRA-small (Clark et al., 2020), is a transformer-based model. ELECTRA-small trains with the goal of predicting replacement or non-replacement of each token in a corrupted input. ELECTRA-small shares architecture with BERT (Devlin et al., 2018), but compared to training a model to remake replaced input tokens, ELECTRA-small's goal enables comparatively more efficient training. Therefore, ELECTRA-small is a suitable choice for an environment with limited computing resources. The Stanford Natural Language Inference (SNLI) corpus (Bowman et al., 2015), a dataset of 570,000 sentence pairs, is a commonly used dataset for training models to perform NLI. Prior research highlights that NLI models trained on SNLI can adopt shallow heuristics (Gururangan et al., 2018), (Gubelmann and Handschuh, 2022). The SNLI model specifically contains a disproportionately small set of sentence pairs with lexical overlap, and the examples present do not pose difficult syntactic challenges to the model, such as subject-object swapping. The small number lexical overlap examples present are also disproportionately gold-labeled with entailment.

This report is split into two main sections. The first section shows that the ELECTRA-small model performs adequately on the SNLI development set after training on the SNLI corpus, but struggles with accuracy on sentence pairs with lexical overlap, such as examples in the dataset from McCoy et al. (2019). The first section also presents analysis on the characteristics of the SNLI corpus that contribute to the aforementioned poor performance on sentence pairs with lexical

overlap. The second section details two methods for training on adversarial data: dynamic generation during training and static corpus construction. The section closes with analysis on these two methods' ability to drive better performance by ELECTRA-small on sentence pairs with lexical overlap. I propose that training on adversarial data is an effective way to avoid ELECTRA-small adopting shallow lexical overlap heuristics, but specific care needs to be taken when generating adversarial data for training to include examples of non-entailment gold-label hypotheses that are not prevalent in the SNLI corpus.

2 ELECTRA-small and SNLI: Struggles with Adversarial Attacks

In order to analyze ELECTRA-small's NLI performance to find focus areas to improve, I chose to train using the SNLI corpus. I used 55,0153 of the sentence pairs of the SNLI corpus for training. I used the Google Colab platform with a V100 GPU to conduct training. After 3 epochs of training, evaluation on a held-out development set from the SNLI corpus, denoted as `SNLI-dev`, yielded an evaluation accuracy of 89.23%. Thorough analysis on this result, as well as model performance categorization, follows in the Evaluation Results section.

2.1 Evaluation Results

2.1.1 Evaluation Results Summary

Evaluation Dataset	Size	Correct	Incorrect	Accuracy (%)
SNLI-dev [1]	9842	8782	1060	89.23
glockner-acl18 [2]	8193	7584	609	92.57
mccoy-etal-2019 [3]	30000	15443	14557	51.48

Table 1: Evaluation Performance of ELECTRA-small trained for three epochs on three evaluation datasets. *Correct* refers to the count of examples for the row's dataset that the model correctly predicted the gold-label. *Incorrect* refers to the count of examples for the row's dataset that the model incorrectly predicts the gold-label.

Table 1 shows the three development datasets I used to evaluate the trained model's performance. `SNLI-dev` is the development set held out from the SNLI corpus. `glockner-acl18` and `mccoy-etal-2019` are both adversarial challenge sets. Detailed explanation of these datasets and analysis of the results follows in the following section.

2.1.2 Adversarial Challenge Sets Description

The model's 89.23% evaluation accuracy on `SNLI-dev` was as expected per the project specification. Therefore, to further analyze ELECTRA-small's performance, I chose to evaluate using `glockner-acl18`. This dataset contains examples with sentence pairs that contain no more than one different word, and specifically swaps words in the following categories: antonyms, cardinals, nationalities, drinks, colors, ordinals, countries, rooms, materials, vegetables, instruments, and planets (Glockner et al., 2018b). For example:

> *Premise*: Several women stand on a platform near the yellow line
> *Hypothesis*: Several women stand on a platform near the red line

Glockner et al. (2018) report that numerous NLI models suffer a downgrade in performance on the `glockner-acl18` dataset as compared to the standard SNLI set, but BERT and ELECTRA-small are not included in their analysis. By evaluating the ELECTRA-small model trained on SNLI using `glocker-acl18` , I hoped to determine if the model was able to perform well on a dataset that strains lexical knowledge based on the categories of substitutions summarized above.

Detailed statistics and analysis follows in the Error Categories and Results Discussion sections, but after observing the 92.57% accuracy (Table 1) on `glockner-acl18`, I chose to also evaluate the trained model's performance on `mccoy-etal-2019`. This dataset contains a robust set of adversarial challenges focused on exploiting heuristics based on syntactic properties, namely lexical overlap. McCoy et al. (2019) present BERT's poor performance on this dataset, so I suspected that ELECTRA-small would struggle with the dataset, meaning I could obtain meaningful insight into specific categories of errors the model makes. My suspicions were confirmed, as the model was only 51.48% accurate on this dataset (Table 1). Note that the `mccoy-etal-2019` has only two possible gold labels: entailment and contradiction (no neutral).

[1] (Bowman et al., 2015)
[2] (Glockner et al., 2018a)
[3] (McCoy et al., 2019)

2.1.3 Error Categories

To start my analysis, I used the following definitions of three broad error categories, from which I derived more specific subcategories:

- **False positives** (FPos) = model predicted entailment, gold label is either neutral or contradiction
- **False neutrals** (FNeut) = model predicted neutral, gold label is either entailment or contradiction
- **False negatives** (FNeg) = model predicted contradiction, gold label is either entailment or neutral

In the context of deriving more specific subcategories, I decided to discard using the `glockner-acl18` results centered around swapping one word as described in the Adversarial Challenge Sets Description section because of the model's strong performance on the dataset. Given the main task was to identify flaws in the model and drive improvement, this was an appropriate decision. See Table 13 in the Appendix for full SNLI-trained ELECTRA-small `glockner-acl18` evaluation results.

Results from evaluating ELECTRA-small's performance on `mccoy-etal-2019` provide meaningful insight into syntactic properties the model struggles to encode. Consider the following error subcategories, to branch off the false positives, false neutrals, and false negatives categories, defined using McCoy et al. (2019)'s insights:

- **Lexical overlap** = the hypothesis contains all words also in the premise, not necessarily in the same order; the hypothesis may also contain other words not in the premise
- **Subsequence** = the hypothesis contains all words also in the premise in the same order; the hypothesis may also contain other words not in the premise

From the definitions above, clearly subsequence is a subset of lexical overlap. For the remainder of this report, lexical overlap example counts should be understood to also include subsequence example counts. Note that McCoy et al. (2019) also define a constituent error category, a subset of subsequence and thus also of lexical overlap, that I chose to not include in my analysis to keep the scope of this project manageable.

Referring to Table 1, I chose to focus my error subcategory performance from examples in the `mccoy-etal-2019` dataset because of the model's poor performance on dataset as compared to `SNLI-dev` and `glockner-acl18`. Table 2 shows the performance of ELECTRA-small `mccoy-etal-2019` with respect to the broad error categories {*false positives*, *false neutrals*, *false negatives*} and the error subcategories {*lexical overlap*, *subsequence*}. Table 3 shows the same, but referencing the `SNLI-dev` evaluation dataset. The discrepancies in both subcategory counts and model accuracy between datasets is explored in Results Discussion.

Subcategory	Total Count	Accuracy %	FPos	FNeut	FNeg
Lexical Overlap	10000	53.36	4460	5	199
Subsequence	10000	50.77	4912	0	11

Table 2: `mccoy-etal-2019` ELECTRA-small (SNLI-trained) performance with respect to the error categories and subcategories described in this section. Recall the dataset has no neutral gold labels, and that subsequence is a subset of lexical overlap. Therefore, the count for lexical overlap above denotes examples containing minimally lexical overlap (therefore including all subsequence examples), and the count for subsequence above denotes examples containing minimally subsequence. Also recall that the constituent subcategory defined by McCoy et al. (2019) is not included in this project's analysis, which makes up the different in total dataset size compared to the sum of lexical overlap and subsequence counts. Total Count refers to the total number of examples of the subcategory.

Subcategory	Total Count	Accuracy %	FPos	FNeut	FNeg
Lexical Overlap	321	99.07	0	1	2
Subsequence	28	100	0	0	0

Table 3: `SNLI-dev` ELECTRA-small (SNLI-trained) performance with respect to the error categories and subcategories described in this section. The performance here is compared against performance on `mccoy-etal-2019` in the next section. Total Count refers to the total number of examples of the subcategory.

2.1.4 Results Discussion

The most notable results comparing Table 2 and Table 3 are summarized below:

- The small count of lexical overlap (321) and subsequence (28) examples in `SNLI-dev` compared to the total size of the dataset (9842)
- The model's evaluation accuracy against `SNLI-dev` on lexical overlap (99.07%) and subsequence (100%) examples compared evaluation accuracy against

mccoy-etal-2019 lexical overlap (53.36%) and subsequence (50.77%) examples.

These discrepancies can be explained by the relative syntactic challenges presented by the mccoy-etal-2019 dataset lexical overlap and subsequence examples compared to those of the SNLI corpus.

Dataset	Dataset Size	Subcategory	Gold Label	Gold Label Count
SNLI-train	550153	Lexical Overlap	Entailment	16023
			Neutral	655
			Contradiction	174
		Subsequence	Entailment	1317
			Neutral	58
			Contradiction	22

Table 4: Statistics on SNLI training data's lexical overlap and subsequence examples. Only 3.06% of the training corpus pairs are lexical overlap or subsequence examples. Of lexical overlap examples, only 4.92% are non-entailment gold-labeled. Of subsequence examples, only 5.73% are non-entailment gold-labeled.

Dataset	Subcategory	Gold Label	Gold Label Count	Model Accuracy %
SNLI-dev	Lexical Overlap	Entailment	315	100
		Neutral	1	0
		Contradiction	5	60.00
	Subsequence	Entailment	28	100
		Neutral	0	0
		Contradiction	0	0
mccoy-etal-2019	Lexical Overlap	Entailment	500	96.02
		Neutral	0	n/a
		Contradiction	5000	10.80
	Subsequence	Entailment	5000	99.78
		Neutral	0	n/a
		Contradiction	5000	1.76

Table 5: Counts of lexical overlap and subsequence examples present in the SNLI-dev and mccoy-etal-2019 datasets.

Table 4 shows details of the training SNLI subset with respect to lexical overlap and subsequence examples. Clearly, the training corpus has a disproportionately small amount, 3.06% of the set, of lexical overlap or subsequence sentence pairs. Within the small number of lexical overlap and subsequence examples, the number of non-entailment gold-labels are near negligible; only 0.165% of the total training corpus are non-entailment gold-labeled lexical overlap or subsequence sentence pairs.

Table 5 shows that the lexical overlap and subsequence examples in SNLI-dev are also disproportionately gold-labeled with entailment compared to lexical overlap and subsequence examples in mccoy-etal-2019. Clearly, ELECTRA-small struggled significantly with both lexical overlap and subsequence examples that had gold labels other than entailment. SNLI-dev contained only 6 such examples across both subcategories out of its total 9842 sentence pairs. Manual examination of the SNLI-dev lexical overlap and subsequence examples shows that the vast majority have hypotheses that effectively condense and summarize the premise, such as the following examples:

Premise: A man selling donuts to a customer during a world exhibition event held in the city of Angeles
Hypothesis: A man selling donuts to a customer
lexical overlap, no subsequence, gold label entailment, model predicted entailment

Premise: A young boy in a field of flowers carrying a ball
Hypothesis: boy in field
lexical overlap, no subsequence, gold label entailment, model predicted entailment

Comparatively, mccoy-etal-2019 lexical overlap and subsequence examples contain challenging syntactic properties, such as subject-object swaps and passive phrasing:

Premise: The president advised the doctor
Hypothesis: The doctor advised the president
lexical overlap, no subsequence, subject-object swap, gold label contradiction, model predicted entailment

Premise: The managers were advised by the athlete
Hypothesis: The managers advised the athlete
lexical overlap, no subsequence, passive phrasing, gold label contradiction, model predicted entailment

Numerous other challenging syntactic property examples are present in mccoy-etal-2019. Refer to Table 14 in the Appendix for full statistics on SNLI-trained ELECTRA-small's performance across all syntactic properties defined by McCoy et al. (2019).

Combining the data in Tables 4 and 5 and the analysis above, it follows that ELECTRA-small struggled to perform on the mccoy-etal-2019 dataset considering the SNLI corpus' (both training and development) lack of challenging lexical overlap and subsequence examples. The literature supports the finding that NLI models trained on SNLI struggle on examples with gold labels other than entailment that contain lexical overlap between the premise and hypothesis. Rajee et. al (2022) present results suggesting BERT adopts heuristics biased towards predicting entailment for sentence pairs containing full word overlap (i.e.

lexical overlap). This concept is well supported by other research (Zhou and Bansal, 2020), (Mendelson and Belinkov, 2021). Specifically, the aforementioned literature notes that transformer-based models, despite general conceptions of their robustness, can still adopt heuristics, such as those categorized above involving lexical overlap, when trained to an NLI task. Specifically, transformer-based models are likely to predict entailment between sentence pairs in examples with lexical overlap, and are likely to predict contradiction between sentence pairs in examples without lexical overlap (Naik et al., 2018), (Liu et al., 2022); however, literature also suggests augmenting training datasets with adversarial examples can lead to increased transformer-based model performance on lexical overlap examples (McCoy et al., 2019).

Equipped with the results shown in this section, literature support, and knowledge of the shared transformer-based architecture between ELECTRA-small and BERT, I decided to focus my efforts on improving ELECTRA-small's performance on the challenging lexical overlap and subsequence examples present in the `mccoy-etal-2019` dataset. The results suggesting training dataset manipulation can lead to increase performance are not surprising, considering joint-direction transformer architectures are able to capture interactions between two sentences; that is, I chose to focus efforts on enhancing training datasets as opposed to tweaking ELECTRA-small's architecture. In order to keep the scope of this effort manageable, I chose to focus my improvement efforts related strictly to `mccoy-etal-2019`, not the other errors shown from evaluation against `SNLI-dev` and `glockner-acl18`. The results shown in this section support this choice, as there is much more room for improvement with respect to my chosen focus.

3 Training Electra-small on Adversarial Data

To improve ELECTRA-small's performance against the challenging lexical overlap and subsequence examples present in the `mccoy-etal-2019` dataset, I chose to train the model on adversarial data. Specifically, I attempted two methods with the aim of driving ELECTRA-small to encode syntactic properties rather than adopt heuristics based on lexical overlap in sentence pairs:

- Generating adversarial samples from the SNLI training set while training to inject into the training set
- Augmenting SNLI training set with a subset of `mccoy-etal-2019` data prior to training

3.1 Generating Adversarial Samples

In order to generate adversarial samples from the SNLI training set while training to inject into the training set, I utilized the TextAttack framework (Morris et al., 2020). Given a predefined model such as ELECTRA-small, and a baseline training set such as the SNLI corpus, TextAttack allows generation of adversarial samples from the baseline training set (between epochs) during training to add to the data looped over during training epochs. The framework allows specification of the method to generate the adversarial examples. I chose to use the TEXTFOOLER (Jin et al., 2019) method. This method generates adversarial examples by selecting important words in the hypothesis to replace with syntactically similar and grammatically correct alternatives; refer to Jin et. al (2020) for the definition of important words. By using this method, I hoped to divert ELECTRA-small's affinity for lexical overlap heuristics, as by changing words in the hypothesis, overlap between the premise and the hypothesis would decrease.

TEXTFOOLER allows specification of the number of adversarial examples to generate, as well as the specific epochs in which to generate the examples. The environment and parameters I used are summarized below:

- Two separate training runs, one of which generating 5000 adversarial examples out of the greater than 560000 possible from the SNLI training set, and the other generating 50000. For both of these training runs, the following remained consistent:
  - ELECTRA-small model with baseline SNLI training set to start
  - Trained on Google Colab platform with a V100 GPU
  - Adam Optimization for learning rate
  - 3 epochs of training, 1 clean epoch to start training
  - Attack interval of 1 epoch, meaning adversarial examples (denoted as adversarials for the remainder of this report) were generated between the 1st and 2nd epoch for use in the 2nd epoch, as well as between the 2nd and 3rd epoch for use in the 3rd epoch

3.1.1 Effectiveness

Adversarials Generated	Dataset	Dataset Size	Evaluation Accuracy (%)
5000	**SNLI-dev**	9842	89.11
	glockner-acl18	8193	94.48
	mccoy-etal-2019	30000	52.01
50000	**SNLI-dev**	9842	88.80
	glockner-acl18	8193	94.96
	mccoy-etal-2019	30000	50.70

Table 6: Evaluation Performance of ELECTRA-small trained for 3 epochs using TEXTFOOLER; both training runs are shown, the first of which with 5000 adversarials generated between epochs 2 and 3, and the second with 50000 adversarials generated between epochs 2 and 3.

Adversarials Generated	Subcategory	Subcategory Count	Accuracy %	FPos	FNeut	FNeg
5000	**Lexical Overlap**	10000	57.00	4117	0	183
	Subsequence	10000	50.35	4957	0	8
50000	**Lexical Overlap**	10000	51.93	4714	0	93
	Subsequence	10000	50.37	4963	0	0

Table 7: `mccoy-etal-2019` ELECTRA-small (SNLI-trained with TEXTFOOLER generation) performance with respect to lexical overlap and subsequence examples. Recall the dataset has no neutral gold labels, and that subsequence is a subset of lexical overlap. Refer to Tables 15 and 16 in the Appendix for full statistics on this ELECTRA-small model's performance across all syntactic properties defined by McCoy et al. (2019).

Adversarials Generated	Subcategory	Subcategory Count	Accuracy %	FPos	FNeut	FNeg
5000	**Lexical Overlap**	321	99.07	0	1	2
	Subsequence	28	100	0	0	0
50000	**Lexical Overlap**	321	98.44	1	1	3
	Subsequence	28	100	0	0	0

Table 8: `SNLI-dev` ELECTRA-small (SNLI-trained with TEXTFOOLER generation) performance with respect to lexical overlap and subsequence examples.

Several key observations follow from Tables 6-8:

1. The difference between evaluation accuracy of ELECTRA-small trained on SNLI vs. trained on SNLI with 5000 TEXTFOOLER adversarials generated vs. trained on SNLI with 50000 TEXTFOOLER adversarials generated is marginal across `SNLI-dev`, `glocker-acl18`, and`mcoy-etal-2019`. For all three ELECTRA-small training methods studied:
   (a) The difference in accuracy on `SNLI-dev` is within 0.43%
   (b) The difference in accuracy on `glockner-acl18` is within 2.4%. Both TEXTFOOLER trained models do perform marginally better.
   (c) The difference in accuracy on `mccoy-etal-2019` is within 0.8%.The 5000 advertorials TEXTFOOLER-trained model performs marginally better compared to the normal SNLI-trained model, and the 50000 advertorials TEXTFOOLER trained model performs marginally worse.
2. The difference between evaluation accuracy of ELECTRA-small trained on SNLI vs. trained on SNLI with 5000 TEXTFOOLER adversarials generated vs. trained on SNLI with 50000 TEXTFOOLER adversarials generated is marginal across all lexical overlap and subsequence examples in `SNLI-dev` and `glocker-acl18`. For the aforementioned three ELECTRA-small training methods:
   (a) There is no difference in performance on `SNLI-dev` subsequence examples. The 50000 adversarials TEXTFOOLER trained model performs marginally worse than the SNLI-trained model on lexical overlap examples by 0.63%. There is no difference between the 5000 adversarials TEXTFOOLER-trained model compared to the SNLI-trained model on lexical overlap examples.
   (b) For `mccoy-etal-2019`: The 5000 adversarials TEXTFOOLER-trained model is 3.64% more accurate on lexical overlap examples compared to the SNLI-trained model, and is marginally (0.42%) less accurate on subsequence examples. The 50000 adversarials TEXTFOOLER-trained model performs marginally worse on both lexical overlap and subsequence examples compared to the SNLI-trained model.

In summary, there is marginal performance difference across all three datasets, as well as

specifically on lexical and subsequence examples. This can be explained primarily by the adversarial generation method of TEXTFOOLER. TEXTFOOLER generates syntactically similar and grammatically correct alternative hypotheses, meaning the true gold label does not change. Therefore, the flaw in the SNLI training discussed in the earlier Results Discussion section persists: the lexical overlap and subsequence examples present in the SNLI dataset are disproportionately gold-labeled with entailment and not challenging, and TEXTFOOLER's methods do not change this. Therefore, ELECTRA-small remains prone to use lexical overlap-based heuristics, explaining the marginal difference we observe. Additionally, manual examination of the adversarial examples generated by TEXTFOOLER proved that the vast majority of examples generated only swapped 1 or 2 words, meaning sentence pairs with large amounts of lexical overlap persist, though not necessarily at 100%. This leaves ELECTRA-small still prone to overlap-based heuristics, though it may push training away from fully leaning on 100% overlap-based heuristics.

Lastly, as shown in Table 13 of the Appendix, the SNLI-trained (no TEXTFOOLER) ELECTRA-small successfully predicted 893 of 894 synonyms examples present in the `glockner-acl18` dataset. This follows from ELECTRA-small sharing architecture with BERT, including use of positional and segment embeddings that can capture word similarity. So, in addition to not generating sufficient contradictory challenging lexical overlap and subsequence examples while training, TEXTFOOLER proved ineffective for my goal due to ELECTRA-small's ability to discern word similarity through use of embeddings. Therefore, I decided to shift focus to manually augmenting the SNLI training set with contradictory challenging lexical overlap and subsequence examples by taking a subset of `mccoy-etal-2019` data.

3.2 Augmenting SNLI Training Set

In order to augment the SNLI training corpus with a subset of `mccoy-etal-2019` data prior to training, I generated a training set that combined 60% of the `mccoy-etal-2019` data along with the full SNLI training set, meaning 40% of the `mccoy-etal-2019` data was set aside for performance evaluation. With that 60% training / 40% development split, I evenly sampled across all of the syntactic properties presented by McCoy et al. (2019) as shown in Tables 14-17 in the Appendix. Prior to training, I made sure to shuffle the merged training data. I performed training on a Google Colab platform with a V100 GPU.

3.2.1 Effectiveness

Dataset	Dataset Size	Evaluation Accuracy (%)
SNLI-dev	9842	89.22
glockner-acl18	8193	90.38
40% mccoy-etal-2019	12000	100.00

Table 9: ELECTRA-small(60%-McCoy-augmented-SNLI-trained) performance on the three datasets discussed in section 2.

Subcategory	Subcategory Count	Accuracy %	FPos	FNeut	FNeg
Lexical Overlap	4000	100.00	0	0	0
Subsequence	4000	100.00	0	0	0

Table 10: ELECTRA-small (60%-McCoy-augmented-SNLI-trained) performance on the 40% withheld `mccoy-etal-2019` data with respect to lexical overlap and subsequence examples. Recall the dataset has no neutral gold labels, and that subsequence is a subset of lexical overlap. Refer to Table 17 in the Appendix for full statistics on this ELECTRA-small model's performance across all syntactic properties defined by McCoy et al. (2019).

Subcategory	Subcategory Count	Accuracy %	FPos	FNeut	FNeg
Lexical Overlap	321	99.07	0	1	2
Subsequence	28	100.00	0	0	0

Table 11: ELECTRA-small(60%-McCoy-augmented-SNLI-trained) performance on the `SNLI-dev` data with respect to lexical overlap and subsequence examples.

Several key observations follow from Tables 9-11:

1. ELECTRA-small trained on SNLI augmented with 60% of `mccoy-etal-2019` has the exact same accuracy on the `SNLI-dev` set as ELECTRA-small trained on regular SNLI

2. ELECTRA-small trained on SNLI augmented with 60% of `mccoy-etal-2019` is marginally less accurate, 2.17% on the `glocker-acl18` dataset, compared to ELECTRA-small trained on regular SNLI

3. ELECTRA-small trained on SNLI augmented with 60% of the mccoy-etal-2019 performs astonishingly well on lexical overlap and subsequence examples in SNLI-dev and the remaining 40% of mccoy-etal-2019
   (a) Both models perform equally accurately on the lexical overlap and subsequence examples in SNLI-dev
   (b) ELECTRA-small trained on SNLI augmented with 60% of the mccoy-etal-2019 is 100% accurate on the remaining 40% of mccoy-etal-2019. Recall ELECTRA-small trained on regular SNLI had an accuracy of only 51.48 % on the entire mccoy-etal-2019 dataset.

As a sanity check, Table 12 shows the performance of the baseline ELECTRA-small model trained with the regular SNLI training corpus from section 2 on the 40% withheld mccoy-etal-2019 data. Clearly, the performance is consistent with the baseline model's poor performance on the entire mccoy-etal-2019 set.

Dataset	Size	Evaluation Accuracy (%)
40% mccoy-etal-2019 Total	12000	51.58
40% mccoy-etal-2019 Lexical Overlap	4000	53.64
40% mccoy-etal-2019 Subsequence	4000	50.76

Table 12: SNLI-only-trained ELECTRA-small performance on the 40% withheld mccoy-etal-2019 data from section 3.

In summary, ELECTRA-small trained on SNLI augmented with 60% of the mccoy-etal-2019 performs similarly on SNLI-dev and glocker-acl18, but performs near perfectly on all lexical overlap and subsequence examples present in SNLI-dev and remaining 40% of mccoy-etal-2019, making only 3 mistakes out of the total 8349 examples across both sets. McCoy et al. (2019) reported that BERT trained on a dataset augmented with same examples similar to (but not actually from) mccoy-etal-2019 was able to perform better on mccoy-etal-2019. My results, in addition to the aforementioned McCoy et al. (2019) BERT results, suggest that transformer-based NLI models are in fact able to handle lexical overlap and subsequence examples, so long as the models are trained with appropriate datasets. Specifically, SNLI is not an appropriate dataset, as it has an insufficient number of lexical overlap and subsequence sentence pairs with non-entailment gold labels, leading to the heuristics described above. Transformers are well-understood to capture alignment between words, because self-attention mechanisms allow all words in a phrase to attend to each-other (Vaswani et al., 2017). However, we observe that the properties of transformers and self-attention, such as in ELECTRA-small, do not enable models to overcome lack of challenging examples in training datasets. Without introducing training examples capturing non-entailment gold-labeled lexical overlap and subsequence examples, transformer-based models can still fallback on lexical overlap-based heuristics.

4 Conclusion

In this paper, I show that the transformer-based ELECTRA-small NLI model is vulnerable to adopting lexical overlap-based heuristics if the dataset it is trained on does not contain a sufficient amount of non-entailment gold-labeled examples. In the case where these examples are not common in the training dataset, ELECTRA-small performs poorly on datasets such as mccoy-etal-2019 that deliberately construct challenging lexical overlap and subsequence examples. However, ELECTRA-small demonstrated the ability to disregard such heuristics when exposed to a training set augmented with mccoy-etal-2019 examples. These findings suggest that while SNLI is an expansive, well-understood dataset, for the purpose of training transformer-based NLI models, augmenting the dataset with lexical overlap and subsequence examples can drive better general performance on development datasets containing challenging lexical overlap and subsequence sentence pairs.

References

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

632–642, Lisbon, Portugal. Association for Computational Linguistics.

Kevin Clark, Minh-Thang Luong, Quoc V. Le, and Christopher D. Manning. 2020. ELECTRA: pre-training text encoders as discriminators rather than generators. *CoRR*, abs/2003.10555.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: pre-training of deep bidirectional transformers for language understanding. *CoRR*, abs/1810.04805.

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

Max Glockner, Vered Shwartz, and Yoav Goldberg. 2018b. Breaking NLI systems with sentences that require simple lexical inferences. *CoRR*, abs/1805.02266.

Reto Gubelmann and Siegfried Handschuh. 2022. Uncovering more shallow heuristics: Probing the natural language inference capacities of transformer-based pre-trained language models using syllogistic patterns. *CoRR*, abs/2201.07614.

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, New Orleans, Louisiana. Association for Computational Linguistics.

Di Jin, Zhijing Jin, Joey Tianyi Zhou, and Peter Szolovits. 2019. Is BERT really robust? natural language attack on text classification and entailment. *CoRR*, abs/1907.11932.

Ling Liu, Ishan Jindal, and Yunyao Li. 2022. Is semantic-aware bert more linguistically aware? a case study on natural language inference.

Bill MacCartney and Christopher D. Manning. 2008. Modeling semantic containment and exclusion in natural language inference. In *Proceedings of the 22nd International Conference on Computational Linguistics (Coling 2008)*, pages 521–528, Manchester, UK. Coling 2008 Organizing Committee.

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

Michael Mendelson and Yonatan Belinkov. 2021. Debiasing methods in natural language understanding make bias more accessible. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 1545–1557, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

John X. Morris, Eli Lifland, Jin Yong Yoo, and Yanjun Qi. 2020. Textattack: A framework for adversarial attacks in natural language processing. *CoRR*, abs/2005.05909.

Aakanksha Naik, Abhilasha Ravichander, Norman Sadeh, Carolyn Rose, and Graham Neubig. 2018. 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.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *CoRR*, abs/1706.03762.

Xiang Zhou and Mohit Bansal. 2020. Towards robustifying NLI models against lexical dataset biases. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8759–8771, Online. Association for Computational Linguistics.

A Appendices

Category	Category Count	Accuracy %	FPos	FNeut	FNeg
Antonyms	1147	87.36	120	25	0
Synonyms	894	99.89	0	0	1
Cardinals	759	96.97	6	9	8
Nationalities	755	93.51	35	14	0
Drinks	731	94.12	16	17	10
Antonyms-Wordnet	706	82.29	52	67	6
Colors	699	96.57	13	2	9
Ordinals	663	94.42	29	4	4
Countries	613	97.39	7	9	0
Rooms	595	87.23	25	50	1
Materials	397	98.49	4	1	1
Vegetables	109	56.88	11	31	5
Instruments	65	96.92	2	0	0
Planets	60	75.00	10	5	0
Totals	8193	92.57	330	234	45

Table 13: `glockner-acl18` ELECTRA-small (SNLI-trained) performance across the categories described by Glockner et al. (2018). Recall Category refers to the categorization of the single-word swap between the premise and hypothesis

Category	Category Count	Accuracy %	FPos	FNeut	FNeg
ln subject/object swap	1000	4.20	958	0	0
ln preposition	1000	14.40	852	4	0
ln relative clause	1000	10.60	893	1	0
ln passive	1000	1.00	990	0	0
ln conjunction	1000	23.30	767	0	0
le relative clause	1000	97.30	0	0	27
le around prepositional phrase	1000	100.00	0	0	0
le around relative clause	1000	96.40	0	0	36
le conjunction	1000	90.30	0	0	97
le passive	1000	96.10	0	0	39
Lexical Overlap Total	10000	53.36	4460	5	199
sn NP/S	1000	0.00	1000	0	0
sn PP on subject	1000	2.00	980	0	0
sn relative clause on subject	1000	2.00	980	0	0
sn past participle	1000	0.00	1000	0	0
sn NP/Z	1000	4.80	952	0	0
se conjunction	1000	98.90	0	0	11
se adjective	1000	100.00	0	0	0
se understood object	1000	100.00	0	0	0
se relative clause on obj	1000	100.00	0	0	0
se PP on obj	1000	100.00	0	0	0
Subsequence Total	10000	50.77	4912	0	11
cn embedded under if	1000	8.30	917	0	0
cn after if clause	1000	0.00	1000	0	0
cn embedded under verb	1000	0.20	998	0	0
cn disjunction	1000	8.90	911	0	0
cn adverb	1000	0.00	1000	0	0
ce embedded under since	1000	99.70	0	0	3
ce after since clause	1000	100.00	0	0	0
ce embedded under verb	1000	86.30	0	0	137
ce conjunction	1000	99.60	0	0	4
ce adverb	1000	100.00	0	0	0
Constituent Total	10000	50.30	4829	0	144

Table 14: `mccoy-etal-2019` ELECTRA-small (SNLI-trained) performance across the categories described by McCoy et al. (2019).

Category	Category Count	Accuracy %	FPos	FNeut	FNeg
ln subject/object swap	1000	8.50	915	0	0
ln preposition	1000	22.20	778	0	0
ln relative clause	1000	19.90	801	0	0
ln passive	1000	0.00	1000	0	0
ln conjunction	1000	37.70	623	0	0
le relative clause	1000	94.10	0	0	59
le around prepositional phrase	1000	100.00	0	0	0
le around relative clause	1000	100.00	0	0	0
le conjunction	1000	88.00	0	0	120
le passive	1000	99.60	0	0	4
Lexical Overlap Total	10000	57.00	4117	0	183
sn NP/S	1000	0.10	999	0	0
sn PP on subject	1000	2.50	975	0	0
sn relative clause on subject	1000	1.00	990	0	0
sn past participle	1000	0.60	994	0	0
sn NP/Z	1000	0.10	999	0	0
se conjunction	1000	99.20	0	0	8
se adjective	1000	100.00	0	0	0
se understood object	1000	100.00	0	0	0
se relative clause on obj	1000	100.00	0	0	0
se PP on obj	1000	100.00	0	0	0
Subsequence Total	10000	50.35	4957	0	8
cn embedded under if	1000	2.20	977	1	0
cn after if clause	1000	0.00	1000	0	0
cn embedded under verb	1000	0.00	1000	0	0
cn disjunction	1000	1.30	987	0	0
cn adverb	1000	0.00	1000	0	0
ce embedded under since	1000	98.80	0	0	12
ce after since clause	1000	100.00	0	0	0
ce embedded under verb	1000	84.90	0	0	151
ce conjunction	1000	99.50	0	0	5
ce adverb	1000	100.00	0	0	0
Constituent Total	10000	48.67	4964	1	168

Table 15: `mccoy-etal-2019` ELECTRA-small (SNLI-trained with 5k textfooler textattacks) performance across the categories described by McCoy et al. (2019).

Category	Category Count	Accuracy %	FPos	FNeut	FNeg
ln subject/object swap	1000	1.30	987	0	0
ln preposition	1000	7.80	922	0	0
ln relative clause	1000	2.80	972	0	0
ln passive	1000	0.00	1000	0	0
ln conjunction	1000	16.70	833	0	0
le relative clause	1000	96.00	0	0	40
le around prepositional phrase	1000	100.00	0	0	0
le around relative clause	1000	100.00	0	0	0
le conjunction	1000	94.70	0	0	53
le passive	1000	100.00	0	0	0
Lexical Overlap Total	10000	51.93	4714	0	93
sn NP/S	1000	0.00	1000	0	0
sn PP on subject	1000	2.60	974	0	0
sn relative clause on subject	1000	0.10	999	0	0
sn past participle	1000	1.00	990	0	0
sn NP/Z	1000	0.00	1000	0	0
se conjunction	1000	100.00	0	0	0
se adjective	1000	100.00	0	0	0
se understood object	1000	100.00	0	0	0
se relative clause on obj	1000	100.00	0	0	0
se PP on obj	1000	100.00	0	0	0
Subsequence Total	10000	50.37	4963	0	0
cn embedded under if	1000	0.00	1000	0	0
cn after if clause	1000	0.00	1000	0	0
cn embedded under verb	1000	0.00	1000	0	0
cn disjunction	1000	0.40	996	0	0
cn adverb	1000	0.00	1000	0	0
ce embedded under since	1000	100.00	0	0	0
ce after since clause	1000	100.00	0	0	0
ce embedded under verb	1000	97.60	0	0	24
ce conjunction	1000	100.00	0	0	0
ce adverb	1000	100.00	0	0	0
Constituent Total	10000	49.80	4996	0	24

Table 16: `mccoy-etal-2019` ELECTRA-small (SNLI-trained with 50k textfooler textattacks) performance across the categories described by McCoy et al. (2019).

Category	Category Count	Accuracy %	FPos	FNeut	FNeg
ln subject/object swap	1000	100.00	0	0	0
ln preposition	1000	100.00	0	0	0
ln relative clause	1000	100.00	0	0	0
ln passive	1000	100.00	0	0	0
ln conjunction	1000	100.00	0	0	0
le relative clause	1000	100.00	0	0	0
le around prepositional phrase	1000	100.00	0	0	0
le around relative clause	1000	100.00	0	0	0
le conjunction	1000	100.00	0	0	0
le passive	1000	100.00	0	0	0
Lexical Overlap Total	4000	100.00	0	0	0
sn NP/S	1000	100.00	0	0	0
sn PP on subject	1000	100.00	0	0	0
sn relative clause on subject	1000	100.00	0	0	0
sn past participle	1000	100.00	0	0	0
sn NP/Z	1000	100.00	0	0	0
se conjunction	1000	100.00	0	0	0
se adjective	1000	100.00	0	0	0
se understood object	1000	100.00	0	0	0
se relative clause on obj	1000	100.00	0	0	0
se PP on obj	1000	100.00	0	0	0
Subsequence Total	4000	100.00	0	0	0
cn embedded under if	1000	100.00	0	0	0
cn after if clause	1000	100.00	0	0	0
cn embedded under verb	1000	100.00	0	0	0
cn disjunction	1000	100.00	0	0	0
cn adverb	1000	100.00	0	0	0
ce embedded under since	1000	100.00	0	0	0
ce after since clause	1000	100.00	0	0	0
ce embedded under verb	1000	100.00	0	0	0
ce conjunction	1000	100.00	0	0	0
ce adverb	1000	100.00	0	0	0
Constituent Total	4000	100.00	0	0	0

Table 17: `mccoy-etal-2019` 12k subset ELECTRA-small (SNLI-trained augmented with 18k`mccoy-etal-2019` examples) performance across the categories described by McCoy et al. (2019).