Advancing Adversarial NLI

Erica McEvoy
emcevoy6@gatech.edu

Stanley Kwok
skwok30@gatech.edu

Su Mowll
smowll3@gatech.edu

Jared Contrascere jcontra@gatech.edu

Abstract

The Adversarial Natural Language Inference (ANLI) dataset was created as a dynamic benchmark that promises to remove the need to create additional NLI datasets. In this paper, we attempt to expand upon the ANLI dataset by testing whether certain data enhancement techniques are able to decrease any sort of exploitable bias and, in doing so, increase out-of-distribution generalization. The enhancements include simple heuristic duplication removal, heuristic data containing short premise removal, and a simplified version of the Adversarial Filtering (AF) algorithm. After training multiple state-of-the-art language models with the augmented ANLI training dataset, our results show that simple heuristics are not able to improve generalization accuracy while more complex filtration techniques show greater potential.

1. Introduction/Background/Motivation

Recent progress in Natural Language Understanding (NLU) has led to the rapid development of language models that consistently surpass existing NLU benchmarks. The static nature of these benchmark datasets, combined with the costly nature of curating them, imply that the development of new NLU benchmark datasets struggles to keep up with the pace of model improvements. The rapid speed at which these NLU benchmarks become obsolete has therefore raised questions concerning their longevity: Can benchmarks last longer?

Additionally, a growing body of research provides evidence that these language models have learned to exploit biases within the data, and are vulnerable to adversaries [16]. Because the performance of such models considerably degrades on adversarial and out-of-distribution samples (i.e., data encountered "in the wild"), questions have also been raised concerning their robustness: Are these models actually as good as their performance suggests?

Because model brittleness can be easily exposed by both researchers and non-experts, it is currently believed that model performance on traditional NLU benchmarks is overestimated. Spurious biases – defined as unintended correlations between inputs and outputs – found within the benchmark datasets themselves are therefore a major impediment to the generalizability of these language models and so the concept of General NLU is far from achieved despite their high-performance.

The Adversarial Natural Language Inference dataset (ANLI) was created as a way to address the longevity and robustness issues of NLU benchmarks herein described, and was designed to be more challenging to the current state-of-the-art models [18]. The dataset was generated using an iterative, adversarial human-and-model in-the-loop procedure, repeated over 3 rounds. The generation procedure began with a concatenation of the SNLI and MNLI static benchmarks (Natural Language Inference (NLI) is considered the most canonical task within NLU. SNLI and MNLI are two standard benchmark datasets used for NLI). [5, 23]. These datasets consist of a series of sentence pairs (premise and hypothesis statements), and the task seeks to determine an inference relation between them by classifying their relationship as entailment, contradiction, or neural, see Table 1. These benchmarks were initially used to train a BERT-Large [10] model that was utilized to generate predictions on a test set. In the first round, the test set was created by scraping sentences from Wikipedia articles to define new premise statements. Human annotators then manually created corresponding hypothesis statements such that the model would predict the label incorrectly. This defines the basic procedure for Round 1, which is repeated in the subsequent rounds. During Round 2, the premises and annotated hypotheses curated from the previous round were added to the training set. New premise statements were also added to this round of training data from the NLI-Fever [21] benchmark to expand the variety of text. A more powerful language model (RoBERTa [15]) was then trained to generate the next round of predictions. Annotators continued their attempts at generating hypothesis statements that fooled this second model into misclassification. Finally, in Round 3 (R3), a more complex model (RoBERTa ensemble) was trained on additional premises extracted from additional sources (Common Crawl, StoryCloze, etc. [17]). These new sources were identified as a means to increase the diversity of the premises in this final round of training.

With this method, the model became increasingly robust while the test sets became more difficult. It was then demonstrated that models trained on the third round of this new benchmark dataset yielded state-of-the-art performance on a variety of popular NLI tasks, while also maintaining a more difficult test set [18]. A key property of this data generation method is that it can be re-applied in a ”never-ending learning scenario” – the addition of new rounds to the iteration procedure that improve model performance allows for quicker development of stronger benchmarks. In essence, the adversarial data generation procedure can now serve as a *dynamic* benchmark for NLU, thereby prolonging the time for the next generation of language models to surpass.

The ANLI dataset provides many opportunities for further study. Inference labels were provided for the development set, which allows for the possibility of making finer-grained analysis of the NLI model performance. Another direction involves improving the robustness of the model through validation of the training dataset – the examples contained in the development and test sets were properly validated, but not so for the training data.

In this project, a number of filtration methods were investigated and applied to the R3 training data of the ANLI dataset in an attempt to increase performance on the development and test sets, thereby improving the robustness of the model.

Improving the robustness of language models is a critically important issue within the field of NLU. It is well-known that evaluation for many NLU and NLI tasks are broken, due to unreliable and biased datasets used to train such models [6]. For example, one of the first question answering benchmarks (VQA benchmark) was found to have a bias where “2” was the correct answer to 39% of all questions beginning with the phrase “how many” [4]. As long as exploitable and spurious biases are rife within NLU benchmarks, true linguistic reasoning cannot be achieved by the current learning paradigm [19]. The work in this project will check for biases in the ANLI training dataset and attempt to remove them. If successful, the filtration methods in this work will create a stronger ANLI benchmark dataset for future language models to learn from. The hope is that lowering the bias and creating a truly difficult benchmark dataset will allow future models to generalize better to out-of-distribution samples.

To date, the only modification to the ANLI dataset detailed so far in the literature has been to add annotations to help classify and understand the types of data that affects model performance the most [24]. A potentially critical limitation to this work fundamentally lies within its approach. A recent paper by researchers at Google Brain criticised the method of developing adversarially-constructed benchmarks [6]. They maintain that this approach fails to meaningfully address the causes of the robustness and longevity issues discussed herein, and essentially obscures the real issues at hand as well as the abilities of what NLU benchmarks should really be measuring.

2. Approach

Three experiments were conducted to investigate the effectiveness of different data enhancement techniques. In the first experiment, a heuristic filtration method was implemented through the detection of duplicate data entries that were identified and subsequently removed from the training set (see examples provided in Table 1). The presence and volume of duplicate data entries in the ANLI training set had yet to be assessed, and it had been postulated that an overabundance of duplication would imbalance the dataset and effectively reduce the difficulty level of the training dataset for the prediction task at hand. Once corrected, three models were retrained on this new dataset (DistilBERT, BART, and XLNet-Large) and their performance was compared to the evaluated Baseline performance (defined as training the models absent of modifications to the training data) [1].

The second experiment involved implementing a different heuristic filtration method, involving the removal of data containing short premises. The short-premise heuristic was implemented as a proxy for difficulty – as the information density of relatively short premises is so low due to their short lengths, judgements are therefore easier to evaluate their truth against. Any premise shorter than 250 characters was removed along with all associated hypotheses (see examples in Table 1). By removing simple examples, the more difficult examples would have better representation leading to hopefully better out-of-distribution generalization. Once filtered, this training set was used as input to train and compare performance across the same three models in the first experiment.

The goal of the third experiment was to run the training dataset through the AFLite algorithm [7]. The AFLite algorithm is a model that adversarially filters out dataset biases and is broadly applicable to a wide variety of data benchmark types. AFLite has proven to be effective at filtering datasets to improve model generalization accuracy especially for testing on out-of-domain or out-of-distribution datasets, even though the training accuracy will be lowered. However, due to the lack of readily-generalizable open source code for AFLite [2] (it was difficult to adapt the source code examples applied on structured data to generalize to unstructured ANLI data) , an alternative approximation method was instead devised and used. A pre-trained, low-performing model from HuggingFace (“NLI-

Premise	Hypothesis	Label	Experiment
Hausmania is a self-governed cultural house in Oslo, Norway. It is run by a group of underground artists based on collectivist ideology. It is located in Hausmannsgate 34 in Oslo and is a fertile ground for avant-garde art. Also, next door it contains the experimental music venue Kafe Hærverk	Hausmania is self-governed.	e	No Duplicates
Hausmania is a self-governed cultural house in Oslo, Norway. It is run by a group of underground artists based on collectivist ideology. It is located in Hausmannsgate 34 in Oslo and is a fertile ground for avant-garde art. Also, next door it contains the experimental music venue Kafe Hærverk	Hausmania is self-governed.	e	No Duplicates
Romania currently supplies one of the world's largest contingents of troops in Iraq, with nearly 1000 people.	There are at least five Romanians in Iraq	e	No Shorts
Scripps Memorial Hospital Encinitas emergency room doctors and nurses treat two to three injured surfers.	Scripps Memorial Hospital Encinitas can only serve one patient per day.	c	No Shorts
"Automation" is a politically dirty word right now in the U.S. But in China, it's a buzzword that politicians love to repeat. China is installing more robots than any other country, and it's a top priority for the Chinese government. WBUR's Asma Khalid (@asmamk) traveled to the country for more on what this automation revolution looks like.	The Chinese government wants to employ more robot workers.	e	No Easy
Cricket-No balls, wides reign as Bangladesh bowler protests umpiring DHAKA, April 12 A Bangladesh club cricketer peppered the field with no balls and wides to concede 92 runs in four legal deliveries in an extraordinary protest against poor umpiring in the Dhaka Second Division Cricket League, local media reported on Wednesday.	Bangladesh is a place full of balls	n	No Easy

Table 1. Examples of ANLI training data. The first two rows are examples of duplicate entries, the second two samples have short premise statements that were excluded in Experiment 2, and the final two examples were filtered out with the weak model in Experiment 3.

DeBERTa-v3-XSmall" [3, 12]) was utilized and a CrossEncoder [3] model was applied to it. (A CrossEncoder is a model encoding technique where sentence pairs are encoded together, passed into the Transformer network, and a similarity score between the sentence pair is outputted [22].) These are pre-trained models that can be directly used to classify the training data. Any correctly classified samples were removed as they were too easily predicted by the weak model. About 80,000 samples were correctly predicted by the weak model, see Table 1. This approach essentially reproduces the AFLite algorithm "in spirit" but is significantly easier to implement. The remaining training data was then used as input to the same three models as the other two experiments.

A majority of the problems faced during development related to data wrangling due to the complexity inherited from language itself, as well as infrastructure setup. The ANLI dataset and NLI datasets in general are more difficult than other text classification tasks such as sentiment analysis or general classification because it involves multiple sentences that need to be encoded together with special emphasis placed on the single sentence hypothesis compared with the possibly paragraph-long premise. Additionally, many existing models accepted data in different forms and encodings that affected model prediction. Due to the nature of the problem as well, traditional approaches hold the dataset steady while the model is modified such that the general pipeline only needs to be created once and then hyperparameters are just values to be changed. In contrast, dataset modification means that it must be checked against multiple models to ensure that a consistent effect can be detected. This means setting up multiple models and pipelines in a way that can be compared with each other and then multiplying each model by the number of experiments run. This can lead to large training times simply due to the number of models that need to be trained.

One final issue faced was the difficulty in implementing the algorithms in some papers such as AFLite [7]. The high difficulty level of the ANLI data is what makes ANLI stand out for NLI research. Prior filtering is too expensive for training data since it is almost 100 times bigger than its validation and test data. An automatic and efficient way to filter ANLI training data will be valuable for future research. The original plan was to try both heuristics (*e.g*. removing duplicates) and the theoretically proven algorithm (AFLite). However, no ready-to-use or open source implementation of AFLite was found to exist, and it was not clear how to transfer it to the context of the problem in this paper. This led to our new approach mentioned above which was to create a simplified version of AFLite. In order to build a simple model that works with ANLI data, we first tried to use the model provided by Bowman *et al.* [5]. However, we encountered great difficulties in organizing ANLI data to the same format as `data.Field` from `torchtext.legacy`. Part of the failure was due to the fact that `torch.legacy` only worked with an older version of `torchtext`. In the end, we decided to employ a pre-trained weak model from HuggingFace to serve as a model for adversarial filtering in Experiment 3 seen below.

3. Experiments and Results

Three experiments were run with one baseline measurement. The experiments involved some form of data cleaning or augmentation with the hypothesis that this would improve generalization accuracy. The modifications are as follows: an experiment to remove any duplicate data points, an experiment to remove short premises, and an experiment to remove "easy" data points.

3.1. Baseline

The baseline measurement for the ANLI dataset was performed using multiple models with the development set results seen in Table 2. Three models were chosen to benchmark the baseline accuracy. The BART model is a transformer-based autoencoder that uses a novel corrupting scheme to add noise to the data for better performance [14]. The DistilBERT model is another transformer model based on the BERT model, but uses knowledge distillation to reduce the number of parameters and increase the speed of predictions [20]. The XLNet-Large model is a autoregressive model that uses a permutation-based language modeling objective to overcome some of the gaps in the BERT model [25]. These models were trained on the R3 training data using AWS EC2 P2 instances with a text notification system to check for completion.

3.2. Removal of Duplicates (Experiment 1)

Our first filtering experiment involved finding entries in the ANLI R3 dataset with duplicate entries with the same premise, hypothesis, and label. We found 98 instances of pairs of duplicated rows in the dataset, see Table 1. We dropped one of the duplicated rows to leave a unique row in the dataset. The filtered data was then used to train DistilBERT, BART, and XLNet-Large models [1] and the accuracy of these models on the development set can be seen in Table 2 as the "No Duplicates" experiment. Compared to the baseline models, this experiment had slightly lower development set accuracy on the DistilBERT and BART models, but higher accuracy when trained with the XLNet-Large model.

3.3. Removal of Short Premises (Experiment 2)

We next examined the character count of the premise statements in the R3 dataset. A histogram of the character count is shown in Figure 1. Here the shortest premise statement is around 100 characters with the longest close to 800 characters. Nie *et al.* suggested that longer premise statements should lead to harder examples [18]. Therefore, we filtered out entries with character counts less than 250 characters, see Table 1, which corresponds with a natural break in the histogram in Figure 1. The remaining dataset comprised 74,225 entries out of the original 100,458 samples. DistilBERT, BART, and XLNet-Large models were then trained on the filtered data [1], and the development set accuracies are given in Table 2 as the "No Shorts" experiment. For this experiment, the development set accuracy is slightly lower than for all of the baseline models.

3.4. Removal of Easy Data Samples (Experiment 3)

Experiment 3 is based on the Adversarial Filtering algorithm [7] but simplified for ease of development. The purpose of the filtering algorithm is to remove the easiest examples in the head of the distribution so that the more difficult examples in the tail of the distribution are better represented (see Table 1, labeled as the "No Easy" experiment). In essence, it's a way to augment the data to increase the difficulty of the examples by removing the easiest examples. In theory, this should increase generalization accuracy while training accuracy is lowered. This paper attempted to do this by using a simpler and weaker model to predict the classification of the ANLI Round 3 training dataset, and then removed those data points from the training data. This was

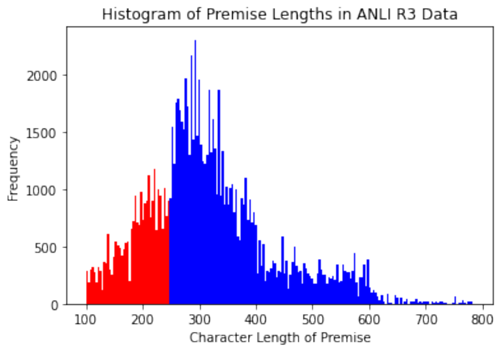

Figure 1. Distribution of character lengths of premise statements of the R3 ANLI training set. The shortest premises (red) were removed from the data, keeping only premises that were of longer duration (blue).

done with the understanding that if a weaker model can correctly predict the data in the ANLI dataset, that data point is too easy. The model employed here is NLI-DeBERTa-v3-XSmall [3, 12], which determines a data point as easy when it is correctly predicted by the model. This model was chosen because it was one of the weakest models found and we did not want a model that was too good where we would have no training data left. This ends up removing roughly 75% of the training data. We trained the three chosen state-of-the-art models [1] with the 25% remaining training data. The results are shown in Table 2 with a slight decrease in test accuracy for the DistilBERT model and slight increases in test accuracy for the BART and XLNet-Large models, compared to the baseline models.

4. Analysis and Discussion

4.1. Experiments

For Experiments 1 and 2, the accuracy results did not generally improve compared to the baseline models, except for Experiment 1 with the XLNet-Large model. The change in the training dataset from baseline to Experiment 1 was minimal (a removal of only 98 samples), but the accuracy changed significantly, if only slightly. This change indicates that duplicate samples could cause a small bias in the data. For experiment 2, none of the models improved the development set accuracy when the short premise statements were removed from the training data. Therefore, the dataset does not appear to biased from the inclusion of short premises.

The results from Experiment 3 were not able to provide full clarity as to whether applying adversarial filtering to the dataset improves out-of-distribution generalization. We present multiple possible theories that could potentially explain the results. For one, adversarial filtering may have little to no predictable effect on model performance [6], in which case, the model results can be explained by the reduced dataset size increasing variance in accuracy numbers amongst the different models. This could potentially be supported by the fact that our baseline results are roughly the same difference from the original paper as they are to our modified results. Another possible answer is that there is in fact a small yet measurable effect, but the DistilBERT model, by it's nature of being distilled, is not able to achieve the same learning performance on downstream tasks as the original BERT model [20], which is generalized by the BART model [14]. Yet another explanation is that the mislabeled data percentage increases after removing easy data. As the difficulty level of premise increases, it becomes increasingly difficult for annotators to come up with an accurate hypothesis based on a given label. In this case, an inappropriate generated hypothesis essentially make the data point become mislabeled. After easy data is removed, mislabel percentage increases as it counts more in difficult data group. Mislabeled data is detrimental to model performance.

4.2. Improvement of Code Quality

Through our research and experimentation with the ANLI dataset, we discovered two bugs in the ANLI code repository [1]. We fixed these and submitted the respective patches to Meta Research for inclusion in the ANLI codebase. These patches correct an issue in loading custom datasets for model fitting and inference [8].

Additionally, we submitted an iPython Notebook (`.ipynb`) file to the ANLI codebase maintainers that accelerates setup and experimentation within Google Colab Pro and Pro+ accounts [9]. This notebook provides researchers a way to quickly set up and run experiments with the ANLI dataset. We have provided this to the ANLI project as an example script.

Our hope is that these contributions will accelerate future research in NLP and support deeper investigation of the ANLI dataset and its effect on large language models. These two contributions enable others to avoid some of the pitfalls we encountered in the course of our project, and we happily provide them to the NLP community to carry this research forward.

4.3. Data Quality

In the context of the Deep Learning ecosystem, the problem we are trying to solve is one related to data quality control. As mentioned before, data quality is crucial to model performance where quality is determined by things such as informational density of data points, level of exploitable biases inherent in the data, and representational power of the dataset distribution in relation to the distribution the model is expected to come across in the wild. Training the model is only half the problem. The second half of the problem comes when you need to determine how good your model

Model	Experiment	Accuracy	Correct	Total
DistilBERT	Baseline	0.442	530	1200
DistilBERT	No Duplicates	0.435	522	1200
DistilBERT	No Shorts	0.430	516	1200
DistilBERT	No Easy	0.417	494	1200
—————	————-	——	——	——
BART	Baseline	0.496	595	1200
BART	No Duplicates	0.493	592	1200
BART	No Shorts	0.473	567	1200
BART	No Easy	0.523	627	1200
—————	————-	——	——	——
XLNet-Large	Baseline	0.497	596	1200
XLNet-Large	No Duplicates	0.509	611	1200
XLNet-Large	No Shorts	0.486	583	1200
XLNet-Large	No Easy	0.513	615	1200

Table 2. Results from Experiments.

is with respect to other models, which is done through the use of benchmarks. Some benchmarks have carried entire fields forward such as the ImageNet dataset or the MNIST dataset [11, 13]. But some of these datasets have flaws that can mislead the field. For example, the ImageNet dataset is texturally biased leading to artificially high accuracy scores and is almost solved in the sense that the best models regularly get 80-90% accuracy [11]. However, because of labeling errors in the ImageNet dataset, it's unclear whether the error can be attributed to the model of the label. This is where ANLI fits into the context of Deep Learning. ANLI is a dataset that can be used as a benchmark where it is hard enough to not be relatively "solved" by models and at the same time, is a good representation of real world human text. This paper attempts to improve upon the quality of data in the ANLI dataset by removing any inherent biases that can be exploited by models to get artificially high accuracy scores. By improving the quality of the dataset, we are able to improve the quality of any model trained or tested on the dataset.

5. Conclusions and Future Work

In this paper, three data enhancement techniques to the ANLI dataset were tested to ascertain whether they increased generalization accuracy. Two were simple heuristics and the third was an adversarial algorithm based on the AFLite algorithm. The simple heuristics demonstrated no measurable generalization benefit while the adversarial algorithm showed mixed results. We provide a possible theoretical basis for our results while leaving open the possibility of further improvement in our methods.

Additional research into this dataset is warranted, and the use of adversarial samples, in general, for model improvement should be evaluated further. While our experiments covered some initial facets of this, future research could investigate other filtration techniques, such as detecting and analyzing groups of similar premise/hypothesis statements using n-gram size or Levenshtein metrics to assess similarity. These facets were not covered in this paper due to computational and time restraints; however, these methods are generally performant and merit further investigation. Additionally, future work would include improvements to the dataset design such that they adhere to the trustworthy benchmark design criteria outlined in the paper by Bowman *et al.* that emphasizes validity of data points, reliable annotation, sufficient statistical power and disincentives for biased models. One such approach would be through the development of an *adversarial competition* to generate healthy benchmarks, wherein the difficulty of collecting valid task examples that are adversarial to each of multiple systems are therein compared [6].

6. Work Division

Erica McEvoy researched papers on this topic, developed code to filter the training data for duplicates and short premises. She designed and ran experiments, aggregated the results, and attempted to reproduce key aspects of AFLite algorithm. She wrote sections of the report and edited the whole report. Su Mowll researched the background of this topic including the source code for AFLite, contributed to the design of experiments, while focusing on experiment 3 implementation. She wrote and edited some sections of the report. Stanley Kwok contributed to the design for experiment 3 and discussion for the other experiments. He used the HuggingFace library to create code to run experiment 3, and wrote and edited sections of the report. Jared Contrascere modified the ANLI codebase for use in Google Colab Pro+, and integrated Google Colab with Google Drive. He set up AWS and Twilio accounts and AWS instances. He ran experiments and collected the results. He wrote some sections of the report.

References

[1] https://github.com/facebookresearch/anli. 2, 4, 5

[2] https://github.com/swabhs/notebooks_for_aflite. 2

[3] https://huggingface.co/docs/transformers/index. 3, 5

[4] Stanislaw Antol, Aishwarya Agrawal, Jiasen Lu, Margaret Mitchell, Dhruv Batra, C Lawrence Zitnick, and Devi Parikh. Vqa: Visual question answering. *Proceedings of the IEEE international conference on computer vision*, pages 2425–2433, 2015. 2

[5] Samuel R Bowman, Gabor Angeli, Christopher Potts, and Christopher D Manning. A large annotated corpus for learning natural language inference. *https://arxiv.org/abs/1508.05326*, 2015. 1, 4

[6] Samuel R. Bowman and George E. Dahl. What will it take to fix benchmarking in natural language understanding? *https://arxiv.org/abs/2104.02145*, 2021. 2, 5, 6

[7] Ronan Le Bras, Swabha Swayamdipta, Chandra Bhagavatula, Rowan Zellers, Mattew E Peters, Ashish Sabharwal, and Yejinn Choi. Adversarial filters of dataset biases. *https://arxiv.org/abs/2002.04108*, 2020. 2, 4

[8] Jared Contrascere, Su Mowll, Stanley Kwok, and Erica McEvoy, 2022. https://github.com/facebookresearch/anli/pull/29. 5

[9] Jared Contrascere, Su Mowll, Stanley Kwok, and Erica McEvoy, 2022. https://github.com/facebookresearch/anli/pull/30. 5

[10] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *https://arxiv.org/abs/1810.04805*, 2018. 1

[11] Robert Geirhos, Patricia Rubisch, Claudio Michaelis, Matthias Bethge, Felix A. Wichmann, and Wieland Brendel. Imagenet-trained cnns are biased towards texture; increasing shape bias improves accuracy and robustness. *https://arxiv.org/abs/1811.12231*, 2019. 6

[12] Pengcheng He, Xiaodong Liu, Jianfeng Gao, and Weizhu Chen. Deberta: Decoding-enhanced bert with disentangled attention. *https://arxiv.org/abs/2006.03654*, 2021. 3, 5

[13] Yann Lecun, Leon Bottou, Yoshua Bengio, and Patrick Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998. 6

[14] Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Ves Stoyanov, and Luke Zettlemoyer. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *https://arxiv.org/abs/1910.13461*, 2019. 4, 5

[15] Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. Roberta: A robustly optimized bert pretraining approach. *https://arxiv.org/abs/1907.11692*, 2019. 1

[16] Tom McCoy, Ellie Pavlick, and Tal Linzen. Right for the wrong reasons: Diagnosing syntactic heuristics in nature language inference. *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3428–3448, 2019. 1

[17] Nasrin Mostafazadeh, Nathanael Chambers, Xiaodong He, Devi Parikh, Dhruv Batra, Lucy Vanderwende, Pushmeet Kohli, and James Allen. A corpus and evaluation framework for deeper understanding of commonsense stories. *https://arxiv.org/abs/1604.01696*, 2016. 2

[18] Yixin Nie, Adina Williams, Emily Dinan, Mohit Bansal, Jason Weston, and Douwe Kiela. Adversarial nli: A new benchmark for natural language understanding. *https://arxiv.org/abs/1910.14599*, 2019. 1, 2, 4

[19] Nazneen Fatema Rajani, Bryan McCann, Caiming Xiong, and Richard Socher. Explain yourself! leveraging language models for commonsense reasoning. *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 4932–4942, 2019. 2

[20] Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *https://arxiv.org/abs/1910.01108*, 2018. 4, 5

[21] James Thorne, Andreas Vlachos, Christos Christodoulopoulos, and Arpit Mittal. Fever: a large-scle dataset for fact extraction and verification. *https://arxiv.org/abs/1803.05355*, 2018. 1

[22] Jesse Vig and Kalai Ramea. Comparison of transfer-learning approaches for response selection in multi-turn conversations. *Workshop on DSTC7*, 2019. 3

[23] Adina Williams, Nikita Nangia, and Samuel R Bowman. A broad-coverage challenge corpus for sentence understanding through inference. *https://arxiv.org/abs/1704.05426*, 2017. 1

[24] Adina Williams, Tristan Thrush, and Douwe Kiela. Anlizing the adversarial natural language inference dataset. *https://arxiv.org/abs/2010.12729*, 2022. 2

[25] Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Ruslan Salakhutdinov, and Quoc V. Le. Xlnet: Generalized autoregressive pretraining for language understanding. *https://arxiv.org/abs/1811.12231*, 2020. 4

Student Name	Contributed Aspects	Details
Erica McEvoy	Researched topic, filtered data, ran experiments, and wrote report.	Researched papers on this topic, developed code to filter the training data for duplicates and short premises. Struggled with AFLite algorithm. Designed and ran experiments, aggregated results. Wrote and edited report.
Su Mowll	Researched papers, set up experiments and wrote report.	Researched the background of this topic. Contributed the design of experiments. Focused on experiment 3 implementation. Wrote and edited some sections of the report.
Stanley Kwok	Designed some experiments, developed code, and wrote report	Contributed to design for experiment 3 and discussion for the others. Used the HuggingFace library to create code to run experiment 3. Wrote and edited report.
Jared Contrascere	Developed code, ran experiments, and wrote report.	Modified ANLI codebase for use in Google Colab Pro+. Integrated Google Colab with Google Drive. Set up AWS and Twilio accounts. Set up AWS instances. Ran experiments and collected results. Wrote some sections of the report.

Table 3. Contributions of team members.