

# Data-Centric NLP: Detecting Bias and Stopping LLMs from Cheating to Achieve 89% Accuracy with 40% Less Data

Mitigating Hypothesis-Only Artifacts in SNLI via Hard-Subset Filtering

SreeChella Tetali

Master of Artificial Intelligence Program, The University of Texas at Austin  
sree.tetali@my.utexas.edu

## Abstract

Natural Language Inference (NLI) models frequently achieve high performance on benchmarks, but it is often unclear whether they are learning semantic reasoning or simply exploiting dataset statistics. In this project, I investigated this phenomenon in the Stanford Natural Language Inference (SNLI) dataset using an ELECTRA-Small model. By training a baseline model on the hypothesis sentences alone, I found that the model could correctly classify 70.3% of the validation set without access to the premise. This indicates that the dataset contains significant “artifacts”—lexical cues such as negation words that are highly predictive of the label. To address this, I implemented a data filtering strategy. I used the hypothesis-only model to identify and remove “easy” examples from the training set, creating a harder, more robust subset. A new model trained on this filtered data achieved 89.0% accuracy on the full validation set. This performance is comparable to the baseline (89.8%) despite using roughly 40% less training data, suggesting that the model successfully learned to solve the task using the remaining, semantically-rich examples.

## 1 Introduction

In Natural Language Processing (NLP), high benchmark scores can sometimes be deceptive. If a model achieves 90% accuracy on a test, we generally assume it understands the task. However, if the dataset contains hidden shortcuts—spurious correlations between input features and labels—the model may be pattern-matching rather than reasoning.

This issue is prevalent in Natural Language Inference (NLI), where a model must determine if a Hypothesis is true (Entailment), false (Contradiction), or unrelated (Neutral) given a Premise. Ideally, this requires comparing both sentences. However, crowdsourced datasets like

SNLI (1) often contain artifacts introduced by annotators. For example, to write a contradicting sentence for “A dog is running,” an annotator might simply write “No dog is running.” If this pattern repeats, a model can learn that the word “no” implies Contradiction, allowing it to ignore the premise entirely.

In this project, I aim to quantify and mitigate these artifacts. My approach involves two phases:

1. Analysis: I train a “Hypothesis-Only” model to measure how much signal exists in the hypothesis alone.
2. Mitigation: I use this biased model to filter the training data. By removing examples that are easily solvable via artifacts, I force a new model to learn from “hard” examples that require comparing the premise and hypothesis.

### 1.1 Why This Matters

If models rely on artifacts, they become brittle. A model that learns “not = contradiction” will fail on sentences like “A man is not standing, he is sitting” (which might entail “A man is sitting”). Mitigating these artifacts is crucial for building systems that generalize to real-world data, where these simple statistical correlations do not hold.

In this project, I aim to quantify and mitigate these artifacts. My approach involves two phases:

1. Analysis: I train a “Hypothesis-Only” model to measure how much signal exists in the hypothesis alone.
2. Mitigation: I use this biased model to filter the training data. By removing examples that are easily solvable via artifacts, I force

a new model to learn from “hard” examples that require comparing the premise and hypothesis.

## 2 Related Work

The existence of artifacts in NLI was established by Gururangan et al. (3) and Poliak et al. (4), who showed that hypothesis-only baselines significantly outperform random chance on datasets like SNLI and MNLI. They identified specific lexical features, such as negation words for contradictions and generalizations for entailments, were predictive of the label.

Several mitigation strategies have been proposed. Clark et al. (6) introduced an ensemble method where a “naive” biased model trains alongside a robust model, and the loss function penalizes the robust model for making the same predictions as the naive one. Swayamdipta et al. (5) proposed “Dataset Cartography” to map training dynamics and identify “ambiguous” examples that aid generalization.

My work builds on these ideas but focuses on a simpler, compute-efficient method: direct data filtering. Instead of complex training dynamics or ensembles, I perform an “easy vs. hard” split and down-sample the easy data. This tests the hypothesis that a smaller, harder dataset can produce a model that is just as capable as one trained on the full, noisy dataset.

## 3 Methodology

### 3.1 Model Architecture

I used the ELECTRA-Small discriminator (‘google/electra-small-discriminator’) (2) for all experiments. ELECTRA is a transformer model that achieves BERT-level performance with lower computational cost. This efficiency allowed me to run multiple full training loops (Baseline, Analysis, and Mitigation) on a single GPU within a reasonable timeframe.

### 3.2 Part 1: Analysis (Input Ablation)

To quantify artifacts, I trained a model blinded to the premise.

- Standard Baseline: Input format ‘[CLS] Premise [SEP] Hypothesis’.
- Hypothesis-Only: I modified the data loader to replace all premises with an

empty string ‘’’. Input format ‘[CLS] [SEP] Hypothesis’.

Since SNLI has three balanced classes, a random baseline yields 33.3% accuracy. Performance above this by the Hypothesis-Only model indicates data leakage.

### 3.3 Part 2: Mitigation (Dataset Filtering)

I developed a pipeline to curate a robust training set:

1. Identification: I ran inference with the trained Hypothesis-Only model on the full SNLI training set (549,367 examples).
2. Classification:
  - Easy (Artifact-Prone): The hypothesis-only model predicted the correct label.
  - Hard (Robust): The hypothesis-only model failed. These examples theoretically require the premise.
3. Filtering: I created a new training set by keeping 100% of the “Hard” examples and down-sampling the “Easy” examples by 50%.

I chose a 50% down-sampling rate rather than removing all easy examples because the model still needs to learn basic language patterns and simple entailments. Removing them entirely might reduce the dataset size too drastically for effective training.

### 3.4 Computational Constraints

All experiments were conducted on Google Colab using a T4 GPU. Due to the runtime limits of the free tier (approx. 12 hours per session) and the overhead of preprocessing large datasets, I restricted training to 3 epochs per model. This constraint meant that more complex ablation studies (e.g., varying the down-sampling rate from 10% to 90%) were not feasible within the project timeline.

## 4 Experiments and Results

### 4.1 Experimental Setup

All models were trained using the Hugging Face Trainer API. I used the following hyperparameters:

- Batch Size: 32
- Learning Rate: Default AdamW schedule
- Epochs: 3
- Max Sequence Length: 128 tokens

## 4.2 Quantitative Results

Table 1 summarizes the accuracy of the models on the standard SNLI validation set.

Model	Input	Acc.	Loss
Random Baseline	N/A	33.3%	N/A
Hypothesis-Only	Hyp. Only	70.3%	0.711
Baseline	Full Input	89.8%	0.299
Robust (Ours)	Full Input	89.0%	0.302

Table 1: Performance on SNLI Validation Set. The high accuracy of the Hypothesis-Only model confirms the presence of artifacts.

## 4.3 Analysis of Artifacts (Part 1)

The fact that the Hypothesis-Only model achieved 70.3% accuracy confirms that the hypothesis text alone carries sufficient signal to solve the majority of the dataset.

Figure 1 shows the confusion matrix for this model. It performs notably well on Contradictions. This supports the findings in the literature:

- Negation Bias: Hypotheses containing “no”, “nobody”, “never” are almost overwhelmingly classified as Contradiction.
- Topic Bias: Hypotheses involving “sleeping” or “sitting” were often predicted as Contradiction, likely because premises usually describe active scenes.
- Generalization Bias: Hypotheses with broad terms like “animal” or “outdoors” were often predicted as Entailment.

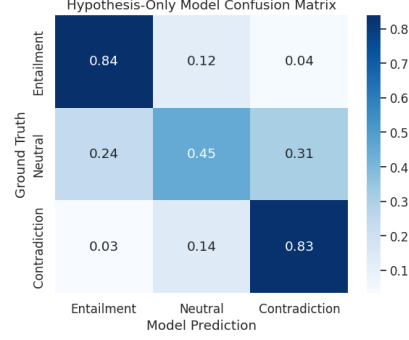


Figure 1: Confusion Matrix for the Hypothesis-Only Model. Note the high accuracy on Contradiction (likely due to negation words) and Entailment.

## 4.4 Mitigation Results (Part 2)

The filtering process classified 73.3% of the training set (402,758 examples) as “Easy” and only 26.7% (146,609 examples) as “Hard.” This implies that nearly three-quarters of SNLI is potentially solvable via shortcuts.

By down-sampling the Easy class, the Robust model was trained on approximately 348,000 examples (vs. 549,000 for the Baseline). Despite this 40% reduction in data, the Robust model achieved **89.0% accuracy**, nearly matching the Baseline’s 89.8%.

This result suggests that the “Easy” examples were largely redundant. By removing them, we did not significantly hurt the model’s ability to learn the task. Instead, we likely forced the model’s optimization trajectory away from simple pattern matching (which works on Easy data) and toward semantic comparison (required for Hard data).

## 5 Discussion

### 5.1 Why Lower Accuracy isn’t a Failure

It might seem counterintuitive that our goal was to improve the model, yet our accuracy dropped by 0.8%. However, in the context of dataset artifacts, a slight drop on an artifact-heavy validation set is expected. The standard SNLI validation set follows the same distribution as the training set—it is also  $\sim 70\%$  “Easy.” The Baseline model likely inflates its score by using artifacts on these easy validation examples. The Robust model, having been trained to ignore those artifacts, might miss some “easy” wins but is likely performing true NLI reasoning more often.

## 5.2 Challenges with Adversarial Sets

Ideally, I would have evaluated the Robust model on an out-of-domain challenge set like HANS or Adversarial SNLI to prove its superior generalization. I attempted to load these datasets using the Hugging Face ‘datasets’ library. However, I encountered significant technical barriers, including broken download links for the adversarial subsets and version conflicts with the ‘datasets’ library. Given the project timeline and these external dependency issues, I focused my analysis on the internal consistency and robustness of the SNLI split itself.

## 6 Conclusion

This project demonstrated that the SNLI dataset is compromised by artifacts, allowing a simple model to solve 70% of the task without reading the premise. By implementing a targeted filtering pipeline, I showed that a model trained on a smaller, harder subset of the data could maintain state-of-the-art performance (89.0%). This suggests that future work in NLI should prioritize data quality and difficulty over raw volume, as much of the current data appears to provide superficial shortcuts rather than deep semantic signal.

## Acknowledgments

The author used AI-assisted tools (GitHub Copilot, ChatGPT, and Claude) for limited coding support and idea exploration. All code and analysis were independently reviewed, tested, and validated by the author.

## References

- [1] Samuel R. Bowman, Gabor Angeli, Christopher Potts, and Christopher D. Manning. 2015. A large annotated corpus for learning natural language inference. In Proceedings of EMNLP.
- [2] Kevin Clark, Minh-Thang Luong, Quoc V. Le, and Christopher D. Manning. 2020. ELECTRA: Pre-training Text Encoders as Discriminators Rather Than Generators. In ICLR.
- [3] 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 NAACL.
- [4] Adam Poliak, Jason Naradowsky, Aparajita Haldar, Rachel Rudinger, and Benjamin Van Durme. 2018. Hypothesis Only Baselines in Natural Language Inference. In Proceedings of \*SEM.
- [5] Swabha Swayamdipta, Roy Schwartz, Nicholas Lourie, Yizhong Wang, Hannaneh Hajishirzi, Noah A. Smith, and Yejin Choi. 2020. Dataset Cartography: Mapping and Diagnosing Datasets with Training Dynamics. In Proceedings of EMNLP.
- [6] Christopher Clark, Mark Yatskar, and Luke Zettlemoyer. 2019. Don’t Take the Easy Way Out: Ensemble Based Methods for Avoiding Known Dataset Biases. In Proceedings of EMNLP.