Deep Data Manipulation to Improve SNLI Models

Abstract

Our goal is to enhance our baseline Natural Language Inference (NLI) model, ELECTRA-small, which was originally trained on the Stanford NLI (SNLI) dataset. To address gaps in model performance, we systematically trained our model on diverse manually created data to improve the performance. By conducting a thorough analysis of our model's performance on the SNLI validation dataset, we identified specific areas where the model struggled to predict labels, revealing inherent gaps. Five labeled datasets were created to cover all the categories where the model had noticeable gaps. Following training on these enhanced datasets, we achieved a substantial increase in accuracy for some categories. Despite the considerable improvements in the accuracy of some specific categories, the overall model accuracy saw a modest enhancement of approximately 1%. Furthermore, We will show why these small modifications will have significant impacts on the model and how datasets play an important role in the model performance.

1 Introduction

Lately, natural language processing (NLP) has been the focus of most studies, with its infinite possibilities and functionalities requiring continuous improvements on the models. This progress has long been measured by standard benchmarks datasets (e.g., Marcus et al., 1993). While these benchmarks were traditionally considered standard, recent studies have highlighted concerns about these evaluation methods. The claim is that datasets may frequently exhibit gaps, introducing potential issues with the reliability of these benchmarks. These gaps can be present due to bias in the data or how the data was created or the intentions behind creating such datasets.

Our main focus for this study will be on the Natural Language Inference (NLI) or recognizing textual entailment (RTE). The focus lies in assessing how a given statement (premise) relates to a proposed hypothesis. We have three possible choices: confirming the hypothesis based on the premise (entailment), contradicting the hypothesis given the premise (contradiction), or indicating that the truth value cannot be ascertained (neutral).

NLI has multiple benchmarks as (Dagan et al., 2006; Giampiccolo et al., 2007; Bowman et al., 2015; Nangia et al., 2017). These benchmarks demonstrated consistently high accuracies, suggesting that these models exhibit proficiency in Natural Language Inference (NLI) across various textual genres. This pattern of success may instill confidence in the effectiveness of these models. However, these models fail on simple examples which lead us to believe that systematic gaps do appear on these benchmarks.

How does one recognize and demonstrate the presence of these gaps, and assess their impact on a model? This paper delves into a comprehensive analysis of model gaps, categorizing them in a generic manner to suit most SNLI models.

Additionally, it is crucial to fix these gaps. Gaps may arise due to two primary factors: model weakness or data weakness, as defined by (Liu et al., 2019). We will explore testing techniques employed to identify these types of gaps and, more specifically, how to handle data weakness.

To assess a model accurately, it is essential to conduct thorough testing. High accuracy on one dataset doesn't guarantee good generalization or performance on similar datasets. Adequate testing and analysis are vital for pinpointing issues and avoiding the time and cost of dataset creation. Crafting a quality dataset of 1000 examples can take about a week for an individual, as suggested by (Gardner et al., 2020).

Through targeted model training based on gap analysis, our objective is to enhance model accuracy by reinforcing its comprehension of negation, numeric concepts, and robust handling of input text. This involves improving the model's performance in scenarios involving spelling mistakes, negation, and numeric data. The creation of labeled datasets, carefully designed to address model gaps, forms a crucial part of this process. Our goal is to accomplish this with the smallest possible dataset size. Achieving this requires extensive testing of both the model and the initial dataset. We plan to incorporate the checklist approach (Ribeiro et al., 2020) and the challenge dataset approach (Liu et al., 2019) to craft an efficient dataset that effectively addresses these identified gaps.

2 Analysis

2.1 Data sources

(Bowman et al., 2015) centered their attention on the SNLI dataset, the same dataset employed in training our model. This dataset was created by Crowdworkers who received a premise sentence and were tasked with formulating a hypothesis sentence. The resulting hypotheses underwent evaluation based on the three categories of entailment relations. Following hypothesis creation, the golden label was assessed by five individuals. A label would be considered gold if at least three of them agreed on its selection.

In the Analysis section, our objective is to illustrate the methods employed to identify gaps in our dataset. We will delve into the testing techniques utilized for categorizing these gaps. Our approach involved a combination of checklists and manual analysis to categorize error types and identify gaps.

2.2 Checklist approach

The Checklist approach entails developing a list of specific aspects that a model should take into account when making predictions. By assessing whether the model has considered each item on the checklist, we can gain insights into the decision-making process of the model.

During our analysis, we considered the following types of checklists: **Vocabulary+POS:** Evaluating whether the model possesses the necessary vocabulary for the task. **Robustness:** Testing the model's resilience to typos, irrelevant changes, or variations in input. **NER (Named Entity Recognition):** Assessing whether the model correctly understands and recognizes named entities in the text. **Negation:** Verifying if the model appropriately handles negations in the language.

After a careful examination of the incorrect examples generated by our initial model, we categorized our errors into these four types mentioned above. We generated 40 examples for each type to assess the model's performance on these instances. Our initial expectation was that the model should not achieve high accuracy on any of these types, as this would contradict the misclassifications observed in the initial model's output. Below are examples that were crafted and evaluated by the model.

Table 1: Checklist Examples.

Error Type	Examples
Vocabulary+ POS	{'premise': "The car is parked in the garage.", 'hypothesis': "The garage is where the car is stored.", 'label': 0},
Robustness	{'premise': "The quick brown fox jumps over the lazy dog.", 'hypothesis': "The quick bown fox jumps over the lazy dog.", 'label': 0}
NER	{'premise': "The Pyramids are in Egypt.", 'hypothesis': "London is where the Pyramids are located.", 'label': 2},
Negation	{'premise': "His comments were not negative;", 'hypothesis': "they were quite positive.", 'label': 0}

Table 1 illustrates one example for each error type. In the Vocabulary+POS example, we aim to test whether the model can correctly recognize that "parked" and "stored" can have the same meaning. In the Robustness example, we deliberately misspelled

"brown" in the hypothesis to assess how well the model handles spelling errors. The NER example examines the model's ability to distinguish between the names "Egypt" and "London." For the Negation example, we evaluate the model's capability to identify negation in the sentence.

After evaluating the model on the generated examples of each type, we observed that results fell short of our expectations. We anticipated a significant variation in the model's performance for each error type. The model exhibited a similar performance across Vocab+POS, and Robustness, averaging around 53% and 50%. The model also demonstrates unsatisfactory performance in NER, showing accuracy around 40%. This highlights our successful identification of model gaps and our capability to identify issues for resolution. On the other hand, and compared to the other categories, Negation showed notably lower performance with an almost 20% difference to Robustness. This shows that our model has difficulty in handling negation examples.

Table 2: Results after evaluation of our initial model on 40 examples of each error type.

Error Type	Accuracy
Robustness	53%
Vocabulary+POS	50%
NER	40%
Negation	33%

2.3 Manual analysis

After evaluating our model on the SNLI validation dataset, we identified 1027 instances of failure. Manually reviewing such a large number of failed examples would be impractical and time-consuming. However, we opted to take a random sample of 100 examples from the 1027 failed instances.

Table 3: Results of 100-samples selected randomly.

Error Type	Percentage
Vocabulary+POS	68%
Robustness	22%
NER	15%
Negation	5%

Subsequently, we attempted to categorize them manually based on the error types outlined in the checklist approach.

We were aware that the numbers derived from the 100-example sample might not provide a robust inference for the entire set of failed examples. Our intention was to verify whether we could effectively observe our category errors in this sample. The high values in the Vocabulary and Robustness categories raised significant questioning. On the contrary, Negation examples were notably scarce. Did the errors we selected based on the previous benchmarks adequately cover all potential error types? The straightforward answer is no, as some of the unidentified failed examples were classified into the nearest category. Our proposed solution is to integrate the Checklist method with the Challenge dataset to yield accurate results and, hopefully, enhance model accuracy.

3 Approach

After conducting a comprehensive analysis, we identified that an effective strategy to address model gaps involves training the model on carefully selected data to fix each error type. Our approach is to integrate the error types identified in our analysis with those outlined in the stress data methodology proposed by (Naik et al., 2018) as shown in table 4.

Table 4: A subset of NLI stress test data proposed by (Naik et al., 2018). Green-highlighted words are added to stress test the model.

Category	Premise	Hypothesis
Word Overlap	Possibly no other country has had such a turbulent history	The country's history has been turbulent **and true is true**.
Negation	Possibly no other country has had such a turbulent history.	The country's history has been turbulent **and false is not true.**
Spelling Errors	I have done what you asked.	I have disobeyed your **ordets**.
Length Mismatch	Possibly no other country has had such a turbulent history and **true is true and true is true and true is true and true is true and true is true.**	The country's history has been turbulent.
Numerical Reasoning	Tim has 350 pounds of cement in 100, 50, and 25 pound bags.	Tim has less than 750 pounds of cement in 100, 50, and 25 pound bags.

The goal is to create datasets for each error type. The concept is straightforward: after training the model on the combined error categories, the model should have three potential outcomes:

Outcome 1: the model adapts well to a challenging test set while maintaining high performance on the original set, suggesting the challenge exposed a lack of diversity in the initial dataset.

Outcome 2: the model's performance remains unchanged on both test sets, indicating a fundamental weakness in adapting to the challenge data distribution.

Outcome 3: the model negatively impacts the original test set performance, signaling a shift towards a challenge distribution that contradicts the original, possibly due to label differences or annotation artifacts.

We discovered that Word Overlap can be merged with Vocab+POS from the earlier checklist method. Negation is identical. Spelling Errors closely resemble Robustness, so we will merge them too. NER Examples will be excluded since our data doesn't include the use of names. The remaining categories will stay separate as they don't share any similarities. We will train our initial model on these distinct datasets and subsequently measure the accuracy of each model individually.

We will create six models, each trained on a different dataset, and compare the results among them:

Model 1: Trained on the original SNLI dataset. **Model 2**: Trained on the original SNLI dataset + 1000 Negation examples. **Model 3**: Trained on the original SNLI dataset + 1000 Spelling Errors examples. **Model 4**: Trained on the original SNLI dataset + 1000 Word Overlap examples. **Model 5**: Trained on the original SNLI dataset + 1000 examples of Length Mismatch. **Model 6**: Trained on the original SNLI dataset + all examples used in Models 2 to 5. Our aim is to achieve positive outcomes (outcome 1), particularly on at least one of the datasets.

4 Results

In this section, we present a comparative analysis of the original (Reference model) and alternative models, including the Negation model, Spelling Errors model, Word Overlap model, Length Mismatch model and Combo model.

The evaluations were carried out on four key sets of data: 10% of each model's dataset, a set of manually created examples (40 examples per type), the SNLI validation dataset (considered as our primary target for evaluating model efficiency), and lastly, we evaluated the model on the negation dataset using the best-performing model identified from the evaluation on SNLI dataset.

4.1 Performance Comparison on 10% of the model dataset

The evaluation was conducted on a subset comprising 10% of each model's dataset. This subset precisely constitutes 10% of the respective datasets associated with each model. To elaborate, the negation model essentially represents the original model, further trained on the 90% subset of the negation dataset, and the remaining 10% for evaluation. This methodology extends uniformly to the rest of the models.

We carefully examined the models' performance based on accuracy. The ensuing visual representation illustrates the comparative accuracy levels between the original model and the alternative models.

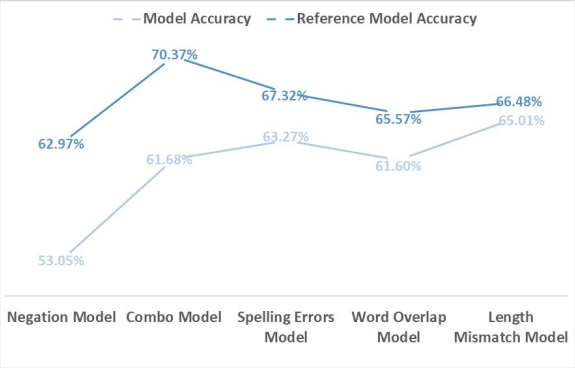

Figure 1: Performance comparison on 10% of the model dataset between the original model and the specific models.

As depicted in figure 1, a noteworthy observation emerges. When each model undergoes training on its respective dataset, a substantial surge in accuracy becomes evident. Particularly, the Negation model exhibits a remarkable enhancement, achieving an almost 10% increase. Conversely, the least notable increase, approximately 1.5%, is observed in the Length Mismatch model.

Figure 2 portrays the relative performance of the reference model in comparison to each alternative model, presented in descending order.

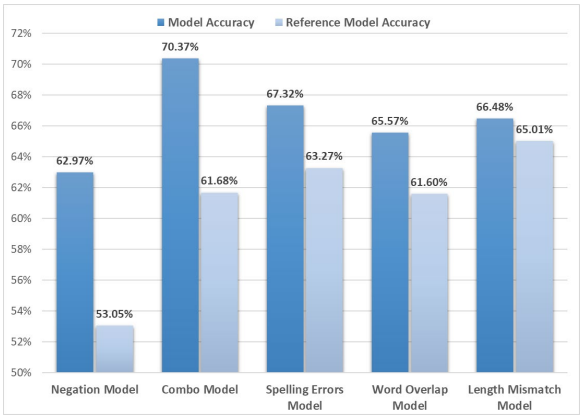

Figure 2: Performance comparison on 10% of the model dataset, showcasing the descending order of specific models based on the difference in accuracy when compared to the original model.

A discernible trend emerges from figure 3, underscoring the remarkable efficacy of the combo model in outperforming both the alternative models and the reference model significantly. Particularly noteworthy is the substantial surge in the Negation model, manifesting an impressive nearly 20% increase relative to the reference model and a 10% augmentation compared to its own previous performance. The Length Mismatch model exhibits a noteworthy performance boost, ascending from 66.5% to 72.2%. Conversely, the impact on the Spelling Errors model is comparatively modest, with a marginal 1.5% increase relative to its prior performance.

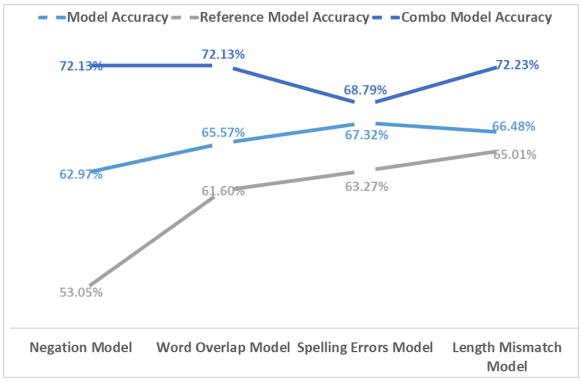

Figure 3: Performance comparison on 10% of the model dataset (excluding the combo's dataset) between the combo model, the original model, and the specific models.

This comprehensive analysis illuminates the nuanced dynamics of each model's performance and highlights the exceptional capabilities of the combo model in the context of accuracy.

4.2 Performance Evaluation on Manually Created Examples

A meticulous evaluation of all models was conducted on manually created examples, the results of which are graphically depicted in the subsequent bar chart (figure 4).

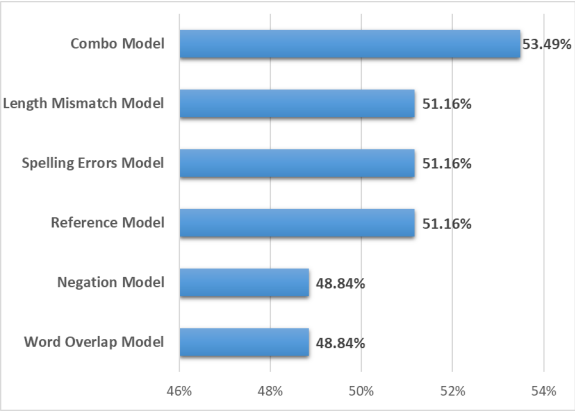

Figure 4: Performance evaluation on manually created set for different models.

Figure 4 shows the discernible pattern within the chart accentuates the superior performance of the combo model, once again surpassing all other individual models. Intriguingly, the evaluation underscores instances where specific models, notably the Negation model and Word Overlap model, were outperformed by the reference model.

This insightful analysis provides valuable perspectives on the models' effectiveness, particularly in the realm of manually curated examples.

4.3 Validation on SNLI Dataset: A Comparative Analysis

In the final part of our evaluation, we carefully analyzed how the models performed on the SNLI validation dataset. The results are visually presented in figure 5.

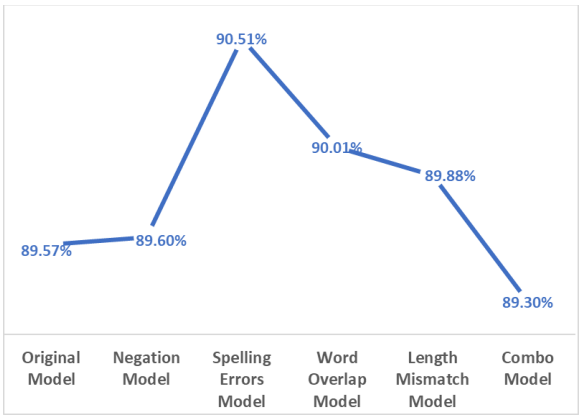

Figure 5: Performance evaluation on SNLI validation set for different models.

Surprisingly, the outcomes across all models displayed a slight difference, where certain models performed slightly better than the original model. The spelling errors model is the best performer among all other models by hitting almost 1% surge compared to the original model. Interestingly, the combo model turned out to be the least effective among the models, although by a small margin, failed to achieve better accuracy than the original model.

After evaluating the accuracy on the SNLI validation set, it becomes evident that the specific models surpassed, not only the original model, but also the combo model. Notably, the accuracy of the spelling errors model exhibited an almost 1% increase compared to the original model.

4.4 Performance Comparison on 10% of the negation dataset

Figure 6 illustrates a bar graph depicting the accuracy of the negation model in comparison to the spelling error model (identified as the top performer in the SNLI dataset) and the original model.

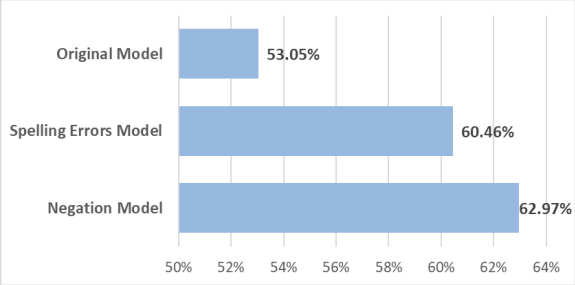

Figure 6: Performance evaluation on 10% of the Negation's set between the original model, the spelling error model, and the negation model.

Analysis of the results reveals that the negation model surpasses both counterparts, by about 10% compared to the original model. Despite this, the spelling errors model exhibits commendable performance, achieving approximately 60.5%. This figure is marginally lower than the negation model by 2.5%, yet significantly surpasses the original model's performance. These results align with expectations, given the negation model's specialized training on a dedicated negation set, contributing to its enhanced performance. Notably, the spelling errors model, despite lacking specific training on the negation set, outperforms the original model and closely approaches the accuracy of the negation model.

5 Conclusion

Our methodology significantly enhanced the NLI model accuracy by effectively identifying and categorizing model gaps. The combined use of checklist and challenging data approaches demonstrated a comprehensive coverage of error categories across various NLI tasks, addressing issues like negation, spelling errors, word overlap, and length mismatch.

Analysis revealed that training the model on the spelling error dataset, in conjunction with the original SNLI dataset, led to the highest improvement in accuracy (about 1%) when evaluated on the target dataset (10,000 SNLI validation examples). Notably, this peak accuracy (around 67%) was achieved when evaluating on 10% test data from created examples for spelling errors, indicating the effectiveness of focused training, compared to 63% for the original model. It also showcases satisfactory performance in specific sets, such as the negation set. Nevertheless, the spelling errors model, while not performing as well in other evaluation sets compared to the combo model, excels in the SNLI set.

The results show that the combo model, which demonstrated outstanding and the best performance across various evaluation sets, compared to the other models, did not exhibit improvement in the target SNLI evaluation set when compared to the original model. It's noteworthy that our combo model (Model 6), trained on all models' created examples (models 2 to 5). This constitutes a major drawback for the combo model, emphasizing the importance of quality over quantity in model performance, highlighting the significance of thorough testing and analysis in enhancing model accuracy by pinpointing effective model gaps.

References

[1] Marcus, M. P., Marcinkiewicz, M. A., & Santorini, B. (1993). Building a Large Annotated Corpus of English: The Penn Treebank. *Computational Linguistics, 19(2), 313-330. DOI: 10.5555/972470.972475.*

[2] Dagan, I., Glickman, O., & Magnini, B. (2006). The PASCAL Recognizing Textual Entailment Challenge. *In Proceedings of the PASCAL Challenges Workshop on Recognising Textual Entailment (pp. 177–190).*

[3] Giampiccolo, D., Magnini, B., Dang, H. T., & Cristianini, N. (2007). The Third PASCAL Recognizing Textual Entailment Challenge. *In Proceedings of the ACL-PASCAL Workshop on Textual Entailment and Paraphrasing (pp. 1–9).*

[4] Bowman, S. R., Angeli, G., Potts, C., & Manning, C. D. (2015). A large annotated corpus for learning natural language inference. *In Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (EMNLP) (pp. 632–642).*

[5] Nangia, N., & Bowman, S. R. (2017). The RepEval 2017 Shared Task: Multi-Genre Natural Language Inference with Sentence Representations. *In Proceedings of the 2nd Workshop on Evaluating Vector-Space Representations for NLP (pp. 1–14).*

[6] Liu, Y., Ott, M., Goyal, N., Du, J., Joshi, M., Chen, D., ... & Stoyanov, V. (2019). Roberta: A robustly optimized BERT approach. *In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the*

9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP) (pp. 1875–1886).

[7] Gardner, M., Grus, J., Neumann, M., Tafjord, O., Liu, Y., Lin, P., ... & Peters, M. (2020). Evaluating models on the Winograd Schema Challenge. *In Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)* (pp. 273–282).

[8] Ribeiro, M. T., Singh, S., & Guestrin, C. (2020). "Why should I trust you?" Explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 1135–1144).

[9] Bowman, S. R., Angeli, G., Potts, C., & Manning, C. D. (2015). A large annotated corpus for learning natural language inference. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing (EMNLP)* (pp. 632–642).

[10] Naik, N., Raskar, R., & Hidalgo, C. A. (2018). Streetscore—Predicting the Perceived Safety of One Million Streetscapes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition* (CVPR) (pp. 812–821).