Dataset and Loss Augmentations for Better Generalization of QA Models: A Dataset Artifact Investigation

Tyler Anderton Michael Huang
University of Texas

Abstract

Question answering is a popular NLP task, driven in part by popular interest in commercializing recent advances in LLMs; however, the excellent performance of these models on common academic QA benchmarks does not always transfer cleanly to industrial contexts (Ribiero et al. 2020). One egregious example of this is when seemingly innocuous changes to the input (e.g a typo or missing word) drastically reduce performance (Gardner et al. 2020). Such model "blind spots" are commonly referred to as dataset artifacts. In this paper we first identify some dataset artifacts that approximate the data imperfections and difficulties that these models might encounter when launched into commercial production. Then we explore methods to mitigate those artifacts during the fine-tuning process of an ELECTRA transformer model on the SQuAD QA benchmark (Clark et al. 2020; Rajpurkar et al. 2016).

1 Introduction

Masked language models have proven themselves to be very powerful at a variety of Natural Language Processing tasks. In 2020, Clark et al. designed a powerful alternative to MLMs in their ELECTRA model. Trained via a method dubbed "replaced token detection" instead of masking, ELECTRA (and even it's smaller version ELECTRA-small, which was selected for use in our investigation) was shown to meet or beat the performance of state-of-the-art MLMs such as BERT, XLNet, and RoBERTa on a variety of tasks, including the popular question-answer dataset SQuAD (Clark et al. 2020).

Designed in 2016, the SQuAD dataset has become the industry standard for evaluating models' performance on single-hop reasoning QA tasks. This reading comprehension dataset includes over 100k questions posed by crowdworkers, with the answer to each question being a segment from an associated Wikipedia article passage (Rajpurkar et al., 2016). The goal of the designers was to create a reading comprehension dataset that was both: (i) large enough to train modern neural models, and (ii) composed of samples that accurately reflect the natural usage characteristics of the English language.

Although SQuAD has been the dataset of choice for many experiments on reading comprehension, it does not come without its drawbacks. For example, the exact answer to the question can always be found word-for-word in the passage (Rajpurkar et al., 2016). This is a convenient way to create a large, natural dataset, but it does not accurately reflect the data found in the real world. In many datasets created from raw real-user data, key words may be missing or misspelled, synonyms or other similar words may be used to refer to the desired response, or misleading or incorrect information can obscure the correct answer (Ribiero et al. 2020).

With ELECTRA and ELECTRA-small having already proved state-of-the-art capabilities on the SQuAD dataset, we wanted to explore opportunities to improve the performance of this architecture even further, not just on the SQuAD dataset, but on modified datasets that are meant to simulate the imperfections and errors a model is likely to encounter in a public environment.

For the purposes of efficiency, we focused on evaluating and improving the ELECTRA-small architecture (here-on referenced as simply 'ELECTRA'), with the assumption that improvements to the small version of ELECTRA should also generalize to the standard ELECTRA architecture.

In order to test ELECTRA's ability to handle imperfect data, we first fine-tuned ELECTRA on the SQuAD training set to create a baseline model. We then made three modifications to the SQuAD dataset for the purpose of testing this baseline model: (i) To imitate missing and misspelled words, we replaced random nouns in the passages of the SQuAD evaluation dataset with gibberish characters before evaluating the baseline model's performance. (ii) To test the baseline model's ability to parse words with similar embeddings, we added a small amount of Gaussian noise to the input embeddings while evaluating performance. (iii) To evaluate the baseline model's ability to find the correct answer amidst misleading or incorrect information, we tested the model's performance on a set of adversarial passages based on the SQuAD dataset (Jia and Liang 2017).

We observed a decrease in the baseline model's performance when evaluated on each of these three modified datasets when compared to the SQuAD evaluation set, indicating room to increase the real-world robustness of our model. By implementing a new loss function and training our model simultaneously on the masked and adversarial versions of the SQuAD data, we observed significant increases in performance, not only on the modified evaluation sets, but also on the unmodified SQuAD evaluation set. This indicates that our training methods successfully increased the durability of our model against flawed input data while maintaining performance on the original benchmark.

2 Methods

This section will describe the three main tests we conducted in the spirit of the CheckList method (Ribiero et al. 2020), and the corresponding improvements we propose to address the corresponding dataset artifacts (*Fig. 1*). As our control, we created a baseline model by fine-tuning for three epochs on the SQuAD dataset. For each of the three testing methods, we evaluated the baseline model performance on the corresponding evaluation set, and found significant drops in performance for each of them. We then implemented methods to mitigate each of these artifacts and combined our methods into two comprehensive models.

2.1 Invariance Testing

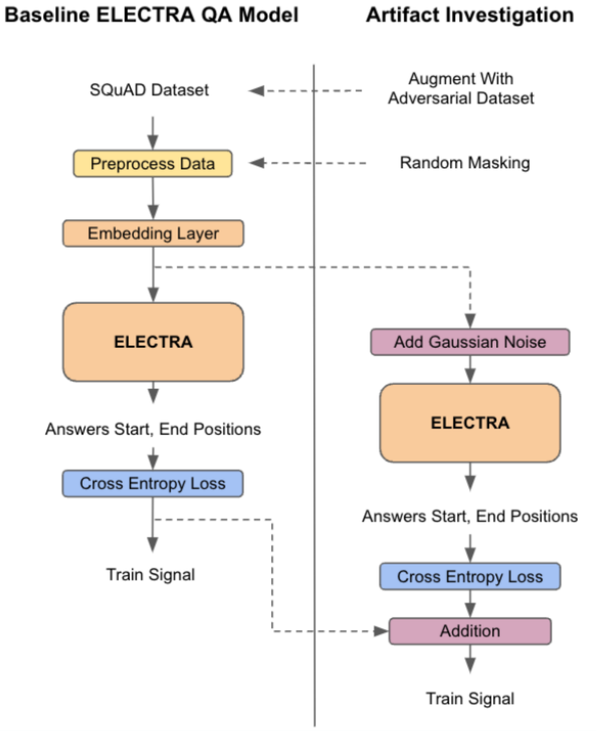


Fig 1. An overview of the data flow for our investigations.

An invariance test applies small label-preserving perturbations to inputs and expects the model predictions to remain the same (Ribiero et al. 2020). In our case, for each input we first sampled noise from normal distributions with a range of variances. Then we added this noise to activations after the ELECTRA embedding layer. We then evaluated the performance of the baseline model on these noised inputs (*Table 2*). The idea is that a well-generalized model should be robust to small variations in the input embedding, which approximate various types of noise such as synonym replacements, typos, etc.

To improve the performance of the baseline model with respect to the Gaussian perturbed evaluation set, we implemented a simple loss function which directly incorporates the new objective into the training signal. Specifically, with the original embedded input *x*, the Gaussian perturbed input *x'*, the model *f*, and original loss function *g*, our new Gaussin Augmented Loss (GAL) is computed as

$$\mathrm{GAL}(x, x') = \gamma \cdot g(f(x)) + (1 - \gamma)g(f(x'))$$

for some constant weight *γ,* which we varied in our experiments. Training with this new loss could be interpreted as roughly doubling the original training set by augmenting each training sample with a corresponding perturbed sample, with the tradeoff of also doubling the training time for the same number of epochs.

2.2 Random Mask Testing

A well-generalized model should be robust to typos or unknown words. To test this, we used NLTKs wordnet to perform POS tagging on the SQuAD examples and selected nouns outside the answer span to replace with gibberish letters (Loper and Bird 2004). We evaluated the performance of the baseline model while varying the number of replacements for each example.

To account for the suspected drop in performance when introducing these masked samples, we experimented with two approaches for augmenting the training set. First, we fine-tuned the pretrained baseline model on the masked training set (using a checkpoint from the baseline training process to maintain the same total number of epochs trained - three). Second, we used the original training procedure, but this time, for each training example, we decided whether or not to introduce masking with a certain probability *p*. For both methods, we experimented with varying the amount of epochs fine-tuning and masking probability *p,* respectively. Both methods expose the model to masked input data during training time to improve, but differ in the distribution of and timing of training on the masked input.

2.3 Adversarial Testing

Finally, we also evaluated our baseline model using samples from a modified extension of the SQuAD dataset. This dataset added over 2500 adversarial examples to the original SQuAD dataset. Each of these adversarial examples was added to the end of the context passage by a crowdworker and written in a way to purposefully and specifically confuse the model from correctly answering the posed question (Jia and Liang 2017). We evaluated our models with a subset of 500 of these adversarial examples.

To improve the baseline models robustness to adversarial inputs, we added the remaining 2060 adversarial samples to the SQuAD training dataset before fine-tuning the ELECTRA model for three epochs.

2.4 Scoring Results

We scored our results for all models and all tests using the standard SQuAD evaluation metrics: exact match and F1. The exact match metric simply measures the percentage of predictions that matches a ground truth answer exactly. F1 measures the percentage of token overlap between the prediction and the answer. The maximum overlap is chosen for each question, and then these scores are averaged across the dataset to yield the total F1 score (Rajpurkar et al., 2016).

3 Results

3.1 Baseline Performance

Our baseline model, trained and evaluated on the original SQuAD dataset, achieved an exact match score of 78.29 and an F1 score of 86.17.

3.2 Gaussian Augmented Loss

The results of the baseline model evaluated on Gaussian perturbed input embeddings are shown in Table 1. The presence of invariance dataset artifacts is evident even at the smallest noise level, already dropping 10 points for both exact match and F1 score. However, the relatively small performance drop with three magnitudes greater noise suggests that most of the artifacts are already exposed by the first noise level. Therefore, we used a variance of 0.0001 for GAL training.

The results of a newly fine-tuned model using GAL with different γ are shown in Table 2. The variation with equal weighting of both losses showed the best performance on perturbed eval set (an improvement of 9 and 8 points in exact match and F1 score, respectively), while improving a point in exact match on the baseline eval set. This suggests that this method even improved generalization with respect to the original evaluation set in addition to resolving the newly discovered dataset artifacts!

3.3 Mask Augmented Training

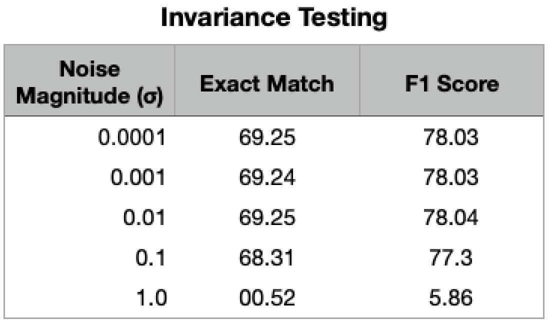
Table 3 shows the performance of the baseline model on the masked dataset. This test induced a greater performance drop than the invariance testing. Again, the relatively similar scores across the number of masked tokens signaled that the majority of the dataset artifacts for this test were exposed even with a single masked noun. We proceeded in the following experiments with three masked tokens. Table 4 displays the results of our two approaches to resolving the masked dataset artifacts. While no strategy improved the performance on the masked dataset to the level of the baseline evaluation set, the approach with the best overall masked and baseline performance was the mask augmented training with *p*=0.25.

3.4 Adversarial Augmented Training

The adversarial dataset yielded the greatest drop in performance when used to evaluate the baseline model, with exact match score dropping from 78.29 to 42.60 and F1 dropping from 86.17 to 48.45 on the original SQuAD and adversarial eval sets respectively. In turn, training on the adversarially augmented dataset also massively improved performance of the baseline model on the adversarial samples, while maintaining nearly identical performance on the original SQuAD eval set. After training with the adversarial samples, our model recovered to near-baseline performance of 73.60 exact match and 78.27 F1 scores.

3.5 Comprehensive Models

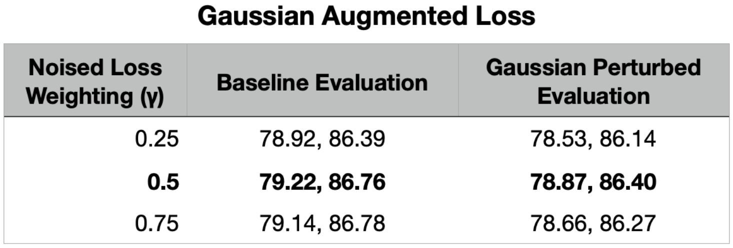
Finally, we created two models using an ensemble of the best performing strategies for mitigating each



Invariance Testing

Noise Magnitude (σ)	Exact Match	F1 Score
0.0001	69.25	78.03
0.001	69.24	78.03
0.01	69.25	78.04
0.1	68.31	77.3
1.0	00.52	5.86

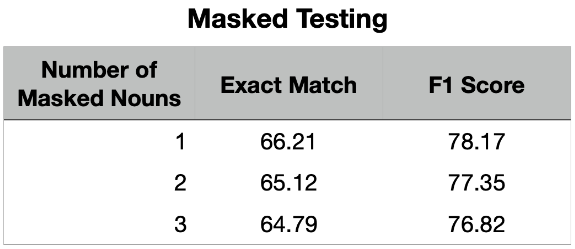
Table 1. Performance of the baseline model on Gaussian perturbed input embeddings, with increasing levels of noise variance.



Gaussian Augmented Loss

Noised Loss Weighting (γ)	Baseline Evaluation	Gaussian Perturbed Evaluation
0.25	78.92, 86.39	78.53, 86.14
0.5	**79.22, 86.76**	**78.87, 86.40**
0.75	79.14, 86.78	78.66, 86.27

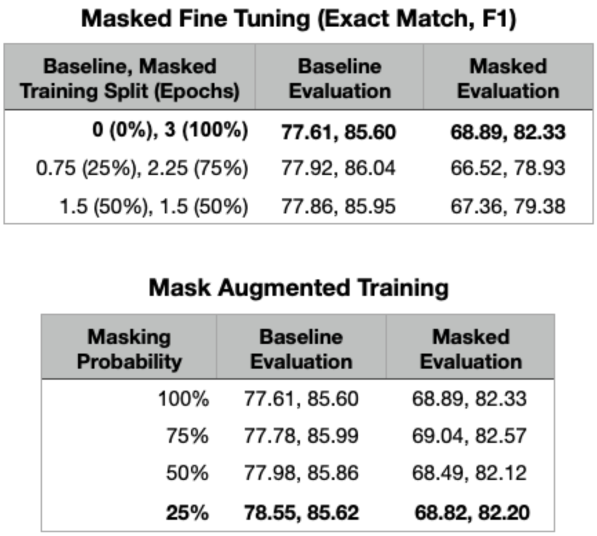
Table 2. Results from retraining the model on SQuAD using GAL, with varying levels of γ. At γ = 1, GAL is entirely dominated by the loss on perturbed input embeddings.



Masked Testing

Number of Masked Nouns	Exact Match	F1 Score
1	66.21	78.17
2	65.12	77.35
3	64.79	76.82

Table 3. Performance of the baseline model on masked data, with increasing numbers of nouns replaced.



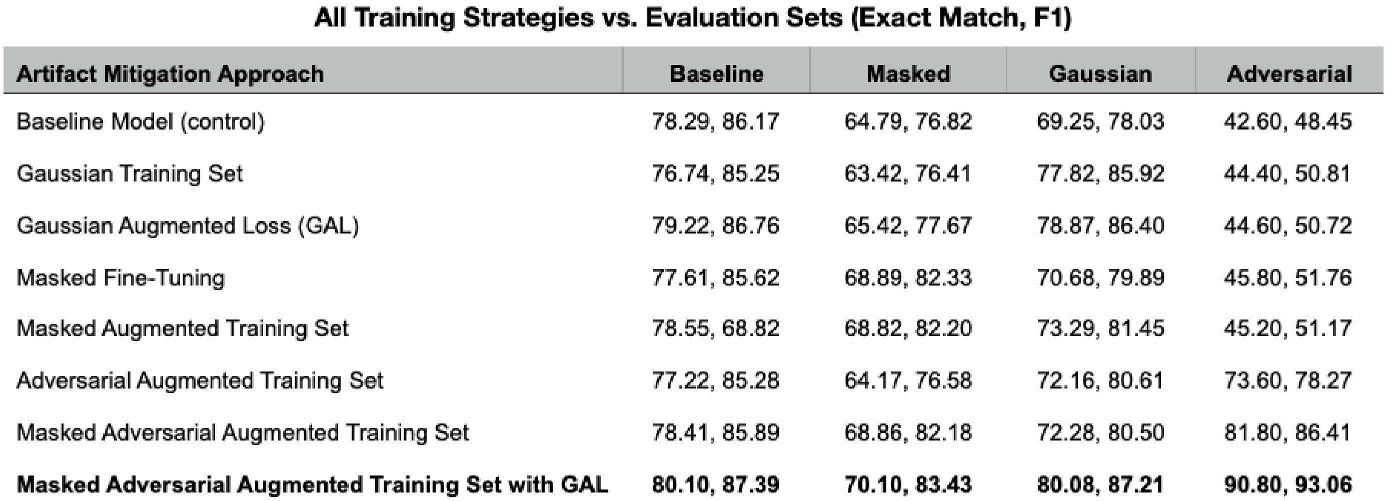
Masked Fine Tuning (Exact Match, F1)

Baseline, Masked Training Split (Epochs)	Baseline Evaluation	Masked Evaluation
0 (0%), 3 (100%)	**77.61, 85.60**	**68.89, 82.33**
0.75 (25%), 2.25 (75%)	77.92, 86.04	66.52, 78.93
1.5 (50%), 1.5 (50%)	77.86, 85.95	67.36, 79.38

Mask Augmented Training

Masking Probability	Baseline Evaluation	Masked Evaluation
100%	77.61, 85.60	68.89, 82.33
75%	77.78, 85.99	69.04, 82.57
50%	77.98, 85.86	68.49, 82.12
25%	**78.55, 85.62**	**68.82, 82.20**

Table 4. Both methods train for a total of 3 epochs. (upper) Results from fine-tuning the baseline model on masked data. (lower) Results from training on the original dataset with a mask probability.



All Training Strategies vs. Evaluation Sets (Exact Match, F1)

Artifact Mitigation Approach	Baseline	Masked	Gaussian	Adversarial
Baseline Model (control)	78.29, 86.17	64.79, 76.82	69.25, 78.03	42.60, 48.45
Gaussian Training Set	76.74, 85.25	63.42, 76.41	77.82, 85.92	44.40, 50.81
Gaussian Augmented Loss (GAL)	79.22, 86.76	65.42, 77.67	78.87, 86.40	44.60, 50.72
Masked Fine-Tuning	77.61, 85.62	68.89, 82.33	70.68, 79.89	45.80, 51.76
Masked Augmented Training Set	78.55, 68.82	68.82, 82.20	73.29, 81.45	45.20, 51.17
Adversarial Augmented Training Set	77.22, 85.28	64.17, 76.58	72.16, 80.61	73.60, 78.27
Masked Adversarial Augmented Training Set	78.41, 85.89	68.86, 82.18	72.28, 80.50	81.80, 86.41
Masked Adversarial Augmented Training Set with GAL	**80.10, 87.39**	**70.10, 83.43**	**80.08, 87.21**	**90.80, 93.06**

Table 5. Results from all improvements attempted, including final ensemble-trained models. Combining the best three approaches yielded the best performance across the board on every evaluation set.

type of dataset artifact. First we fine-tuned a model incorporating only the masked and adversarial data augmentation strategies. Then, we fine-tuned a model using both dataset augmentations and GAL. Table 5 contains the best performance of every improvement strategy attempted. Interestingly, these strategies sometimes improved performance for artifacts which they were not designed to target (e.g. masking augmentation on adversarial set, GAL on masked set). This suggests overlaps between the dataset artifacts uncovered by the different tests.

The results for the two ensemble-trained models are very promising, achieving best performance across the board on all evaluation set variations, including the baseline evaluation set. This means that combining all the strategies is a valid training process for mitigating all the dataset artifacts discovered in this paper. Furthermore, these strategies even improved the generalization of the model with respect to the original evaluation set by two points in exact match and one point in F1 score.

4 Conclusion

Overall we found impressive increases in reading comprehension performance after applying our modifications to the ELECTRA-small model's loss function and training dataset. Even though this "replaced token detection" architecture has already proven state-of-the-art capabilities on a broad range of standard NLP tasks, our findings on these dataset artifacts indicate more work is needed in order to prepare these models for production in commercially available applications (Clark et al. 2020). Our proposals for data and loss augmentation methods yielded significant performance gains on the modified datasets and indicate a significant step toward bolstering NLP models against these common data artifacts. Still, there are additional investigations we propose for further development.

During our masking investigation, we tested the masking of only nouns outside the answer span, which already showed significant room for improvement in the baseline model. While replacing words within the answer span was outside the scope of our investigation, we believe increasing the difficulty of the masking test by various similar means is a promising direction to uncover more dataset artifacts.

One shortcoming of the SQuAD dataset for modeling real-world data is that the correct answer must be present word-for-word in the passage. SQuAD 2.0 is an extension of SQuAD 1.0 that includes an additional 50k+ questions that cannot be answered by the passage. This dataset tests a model's ability to determine whether or not a question is even answerable (Rajpurkar et al. 2018). This ability to deem questions unanswerable may have a significant impact on reading comprehension performance when evaluated on the modified datasets from this paper.

References

K. Clark, M. Luong, Q. Le, C. Manning: ELECTRA: Pre-training Text Encoders as Discriminators Rather Than Generators. In *International Conference on Learning Representations (ICLR)*, 2020.

J. Devlin, M. Chang, K. Lee, K. Toutanova: BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *North American Chapter of the Association for Computational Linguistics (NAACL)*, 2019.

M. Gardner, Y. Artzi, V. Basmova, J. Berant, B. Bogin, S. Chen, P. Dasigi, D. Dua, Y. Elazar, A. Gottumukkala, N. Gupta, H. Hajishirzi, G. Ilharco, D. Khashabi, K. Lin, J. Liu, N. Liu, P. Mulcaire, Q. Ning, S. Singh, N. Smith, S. Subramanian, R. Tsarfaty, E. Wallace, A. Zhang, B. Zhou: Evaluating Models' Local Decision Boundaries via Contrast Sets. In *Findings of the Association for Computational Linguistics*, 2020.

R. Jia, P. Liang: Adversarial Examples for Evaluating Reading Comprehension Systems: In *Empirical Methods in Natural Language Processing (EMNLP)*, 2017.

E. Loper, S. Bird: NLTK: The Natural Language Toolkit. In *Proceedings of the ACL Interactive Poster and Demonstration Sessions*, 2004

P. Rajpurkar, J. Zhang, K. Lopyrev, P. Liang: SQuAD: 100,000+ Questions for Machine Comprehension of Text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, 2016.

P. Rajpurkar, R. Jia, P. Liang: Know What You Don't Know: Unanswerable Questions for SQuAD. In *56th Annual Meeting of the Association for Computational Linguistics (ACL)*, 2018.

M. Ribeiro, T. Wu, C. Guestrin, S. Singh: - Beyond Accuracy: Behavioral Testing of NLP Models with CheckList. In *Association for Computational Linguistics (ACL)*, 2020.