Adversarial Super Bowl
Improving QA Performance with AdvQA

Jonathan Bown

Abstract

This project undertakes an evaluation of the fine-tuning process of the ELECTRA-small model, specifically focusing on its question-answering (QA) capabilities. The primary objective is to identify inherent weakness in the model's QA performance and to understand how different training datasets influence the performance. Through a series of experiments, the project fine-tunes ELECTRA-small using various QA datasets, each trying to achieve better context understanding. The results are evaluated to determine which datasets enhance the model's performance across a multiple categories. Building a more comprehensive QA dataset helps the pre-trained model to better generalize across question variations that are more realistic. Human performance is challenged by augmenting a standard QA training set with adversarial examples in a unique manner.

1 Introduction

Enhancing the performance of pre-trained natural language processing (NLP) models is an important component of generalization, particularly in the context of more specialized tasks. These pre-trained models, including the ELECTRA-small model known for its efficient BERT-like architecture, are typically fine-tuned on datasets like the Stanford Question Answering Dataset (SQuAD) to adapt them from a general to a more specialized function. However, standard metrics like holdout accuracy often overestimate their performance. This research specifically examines how the ELECTRA-small model, after fine-tuning on the SQuAD dataset, handles a variety of question types, including standard holdout, contrast, adversarial context, and checklist examples. The goal is to assess the model's ability to generalize beyond its training data and then improve its performance with data augmentation tailored to improve weak points.

2 Analysis

In order to best understand how the ELECTRA model generalizes with the QA task I first started with exploring existing methods for diagnosing weaknesses with model performance. These included model ablations, the *competency problems* framework, and changing the data with adversarial, contrast, and checklist sets. The *competency problems* framework involves evaluating how different n-gram portions of the text affect the output. *Model ablations* involve altering the model architecture in certain places that expose weakness. While these two approaches are very relevant for QA model evaluation problems, current issues with established frameworks prevented these from yielding any meaningful results. From there, the focus turned to changing the evaluation data in meaningful ways to expose weaknesses with model comprehension.

This analysis is centered around a holdout set of 30 examples from the validation split of the SQuAD dataset that are relevant to the topic of Super Bowl 50. This topic was chosen because of the richness of different question types that appear with the same context. The questions are also easy to change to get questions that challenge the model at its comprehension ability. Question sets were generated using GPT4 from OpenAI. The original questions were given with a prompt to change the questions

to challenge a model on a specific task like NER, numerical reasoning, paraphrasing, etc. These sets of altered evaluation sets are described in the following sections. These methods were evaluated against five models. The fine-tuned models did not have these generated questions as part of their training data.

a) Adversarial Context The first approach used to probe the model was evaluating the trained model on adversarial challenge sets. The simplest approach to our test scheme related to Super Bowl 50 questions was to insert adversarial facts into the context. This is done similarly to the work that was done to develop the SQuAD Adversarial dataset (Jia and Liang, 2017).

I extended this by making four different sentences, each one with a varying degree of similarity to the existing Super Bowl 50 context. The degrees are High Similarity (HS), Moderate Similarity (MS), Low Similarity (LS), and No Similarity (NS).

Below is the Super Bowl 50 context that is included with this topic in the SQuAD dataset

> *'Super Bowl 50 was an American football game to determine the champion of the National Football League (NFL) for the 2015 season. The American Football Conference (AFC) champion Denver Broncos defeated the National Football Conference (NFC) champion Carolina Panthers 24–10 to earn their third Super Bowl title. The game was played on February 7, 2016, at Levi's Stadium in the San Francisco Bay Area at Santa Clara, California. As this was the 50th Super Bowl, the league emphasized the "golden anniversary" with various gold-themed initiatives, as well as temporarily suspending the tradition of naming each Super Bowl game with Roman numerals (under which the game would have been known as "Super Bowl L"), so that the logo could prominently feature the Arabic numerals 50.'*

Here are the different sentences that were added to the context:

High Similarity: "Super Bowl XLIX, held the previous year, was remarkable for its dramatic finish, with the New England Patriots securing a victory over the Seattle Seahawks."

Moderate Similarity: "Super Bowl XXX, celebrated in 1996, saw the Dallas Cowboys achieving a resounding win against the Pittsburgh Steelers, marking their third Super Bowl victory."

Low Similarity: "The first Super Bowl, played in 1967, was not called the Super Bowl at the time, but was later known as Super Bowl I, setting the precedent for future games."

Minimal Similarity: "Super Bowl XXV, famous for its halftime show featuring a performance by New Kids on the Block, took place in 1991, long before the advent of modern halftime extravaganzas."

These sets were carefully designed to probe the model's ability to understand and interpret challenging questions and contexts. Following the methodology of Jia and Liang (2017), I introduced distracting sentences into the context paragraphs, testing the model's resilience to irrelevant or misleading information. Similarly, I incorporated syntactically and semantically tricky questions, akin to those in Wallace et al. (2019), to assess the model's comprehension capabilities. Our approach was further enriched by integrating phenomena highlighted by Glockner et al. (2018) and McCoy et al. (2019), such as lexical ambiguity and structural heuristics, to further challenge the model's interpretative skills. The objective of these adversarial challenge sets was to identify any potential over-reliance on spurious correlations and to gauge the true depth of the model's understanding of natural language, as suggested by Bartolo et al. (2020). This comprehensive approach allowed us to uncover critical insights into the model's operational boundaries and areas for improvement in handling complex, real-world language scenarios.

b) Contrast Sets Contrast sets are designed to test the robustness and comprehensiveness of NLP models, especially in tasks like QA. The concept of contrast sets was introduced by Gardner et al. (2020) as a means to create examples that are minimally different from those in the existing dataset but have different correct answers or classifications. These sets are particularly useful in revealing the limitations of models that otherwise perform well on standard benchmarks.

The essence of contrast sets lies in their ability to expose whether a model's success is due to genuinely understanding the underlying task or merely

exploiting dataset-specific artifacts. By slightly altering the details in a dataset example, one can observe how these changes affect the model's performance, thereby gaining insights into the model's reasoning process.

For our study on the ELECTRA model trained on the SQuAD dataset, I developed contrast sets specifically tailored to the Super Bowl 50 context. These sets include subtly altered questions and contexts that are closely related to the original Super Bowl 50 dataset but differ in key aspects, challenging the model's ability to comprehend and adapt to these variations. Some examples are included below.

```
"Which NFL team represented the AFC at
Super Bowl 50?"

"What day was the game played on?"

"Who won Super Bowl 50?"
```

These questions from the evaluation set were changed to:

```
"Which NFL team came from the AFC at Super
Bowl 50?"

"What day was the Super Bowl initially
planned to be played on?"

"Who did not win Super Bowl 50?"
```

c) Checklist Sets Checklist sets are an approach introduced by Ribeiro et al. (2020), as a more comprehensive testing methodology for NLP models in certain domains. Checklist sets are designed to systematically test a range of linguistic capabilities of models, providing a more comprehensive understanding of their performance and limitations. The Rebeiro paper also introduced a python tool called CheckList meant to provide classes for evaluating models in a more automated way. This tool was explored as part of this analysis but an open issue with the package prevented the use of their code. The application of Checklist sets for this analysis focuses on four key areas: Named Entity Recognition (NER), True/False Questions, Negation, and Numerical Reasoning. Question sets were developed in each case to focus each of the Super Bowl 50 questions as a set of 30 questions.

d) Named Entity Recognition (NER) The NER questions and answers were generated as variations from the original Super Bowl 50 questions and changed them to ask more directly about the entities in the context. This tests the model's ability to accurately identify and classify named entities within the Super Bowl 50 context, an essential aspect of language comprehension.

Examples

```
"Identify the NFL per Bowl 50.",
"Name the city where Super Bowl 50 took
place.",
"Recognize the NFL team that won Super
Bowl 50.",
"Identify the color used to emphasize the
50th anniversary of the Super Bowl.",
"Determine the theme of Super Bowl 50.",
"Specify the day on which the game of
Super Bowl 50 was played.",
"State what the acronym AFC stands for."
```

e) True/False True/False or Yes/No questions are a more advanced capability of QA models. Typically datasets and models are fine-tuned specifically for this type of task (source). Two question sets were actually generated asking for only True and only False responses to get a clearer representation in the results. This tests the model's comprehension and reasoning skills, determining its ability to discern the truthfulness of statements based on the given context.

Examples

```
"True or False: The AFC stands for the
American Football Conference.",
"True or False: Super Bowl 50 emphasized
the golden anniversary.",
"True or False: AFC stands for American
Football Conference.",
"True or False: The Super Bowl 50 was
played on February 7, 2016.",
"True or False: The Denver Broncos won
Super Bowl 50."
```

f) Negation Negation testing with Checklist sets allows us to evaluate how well the model under-

stands sentences with negations, a critical component in comprehending the intended meaning of complex sentences. This is pivotal in assessing the model's grasp of nuances in language.

Examples

```
"For what season was Super Bowl 50
mistakenly believed to determine the NFL
champion?",
"Which team is often wrongly cited as the
winner of Super Bowl 50?",
"Where was Super Bowl 50 falsely rumored
to be held?",
"What is a common misnomer for the NFL
championship game?",
"What 2015 NFL team was incorrectly
announced as the winner of the AFC
playoff?",
"What brand did not air a debut commercial
during Super Bowl 50?"
```

g) Numerical Reasoning Finally, I employ Checklist sets to test the model's numerical reasoning abilities. This involves evaluating how effectively the model can understand and process numerical information presented in various contexts, a vital skill for many real-world NLP applications.

Examples

```
"Over how many days was the Super Bowl
played?",
"How many teams won Super Bowl 50?",
"How many venues did Super Bowl 50 take
place in?",
"How many cities hosted Super Bowl 50?",
"How many Roman Numerals were used for
Super Bowl 50?",
"How many NFL seasons did Super Bowl 50
decide the champion for?",
```

h) Paraphrased Questions Paraphrased Super Bowl 50 questions were generated using GPT-4 via ChatGPT to reformat the question to have a correct answer but different wording. This technique of paraphrasing helps in evaluating the model's ability to understand the meaning of a question better regardless of its linguistic structure. Paraphrasing is a simple yet powerful way to evaluate the performance of a QA model.

Examples

```
'Which NFL team represented the AFC at
Super Bowl 50?',
'Which NFL team represented the NFC at
Super Bowl 50?',
'Where did Super Bowl 50 take place?'
```

The above questions were changed to:

```
'Which team from the AFC played in Super
Bowl 50?',
'Identify the NFC team that participated
in Super Bowl 50.',
'In which location was Super Bowl 50
held?'
```

By integrating these focused areas into our evaluation using Checklist sets, I aim to derive a comprehensive understanding of our model's capabilities and uncover specific areas that may require further refinement.

3 Methodology

The procedure in this project aims to identify problems and investigate solutions by tracking model performance across several key scenarios. To evaluate model performance, standard metrics such as Exact Match and F1 scores, which are well-established for QA data, are utilized on the holdout datasets.

a) Scenarios for Analysis and Solution Development

- **Generate**: OpenAI's ChatGPT (Version 4.0) was used for generating initial sets of variational questions for the original Super Bowl 50 questions from the evaluation set.
- **Baseline:** Evaluate the performance of the ELECTRA-small model on altered holdout sets.
- **Fine-tuned:** Assess the ELECTRA-small model after fine-tuning on the SQuAD dataset, using the same examples.
- **Investigate:** Identify areas of weakness to guide the development of potential solutions.
- **Add Data:** Seek additional data that can expand the model's understanding of QA datasets, thereby enhancing its ability to accurately answer questions.

- **Improve:** Experiment with various training regimes using the expanded dataset to improve the model's performance on the altered data.

b) Fine-Tuning The ELECTRA-small model was first loaded as a pre-trained model into a transformer model pipeline. This model was then trained on the entire train set of the original SQUAD dataset in a notebook with multiple T4 GPUs. The incorporation of GPUs was important because it drastically improved the feasibility of these experiments to run in 4-5 hours rather than days on multi-core CPUs. After this model was trained and saved, it was loaded into a different notebook and evaluated with the different set of perturbed 30 questions about Super Bowl 50. This process was repeated in each fine-tuning scenario. The element that changed throughout the experiment was the training data composition. The Hugging Face `transformers` and `datasets` libraries were instrumental in constructing the model pipelines and in evaluating the outcomes on the different sets of Super Bowl 50 datasets.

c) Experiments In order to improve the performance of the ELECTRA-small fine-tuned on the different evaluation sets several experiments were performed. These experiments were separated by either the type of model that was used such as baseline vs fine-tuned. The fine-tuned experiments used different QA dataset compositions that were iteratively expanded to include larger amounts of SQuAD variations.

d) Baseline The baseline performance data was obtained by loading the pre-trained ELECTRA-small model and evaluating its performance on different Super Bowl 50 altered holdout sets. It was observed that the pre-trained small version could not correctly answer any questions in the evaluation set, as detailed in the results sections.

e) SQuAD The first attempt at improving the QA ability of ELECTRA-small simply involved fine-tuning on the original SQuAD dataset. The 'train' partition was used for fine-tuning and the 'validation' section of that dataset was used for evaluation. Developed by researchers at Stanford University, this dataset comprises an array of context paragraphs extracted from Wikipedia articles. Paired with these context paragraphs are meticulously crafted sets of questions generated by human annotators, covering a wide range of topics. Each question is accompanied by a corresponding answer, wherein the correct response is a segment of text extracted from the context paragraph (Pranav 2016).

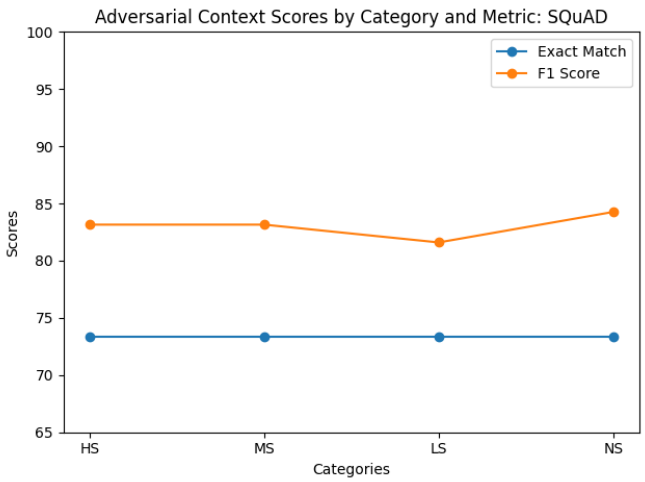
f) SQuAD Adversarial The second attempt to improve model performance involved taking the first experiment further by adding adversarial examples to the original SQuAD dataset. The paper used to discuss adversarial data has an associated dataset called SQuAD Adversarial (Jia and Liang 2017). The model was fine-tuned on the entirety of this dataset. I combined both the 'AddSent' and 'AddOneSent' components of this dataset. The 'AddSent' has up to five candidate adversarial sentences that "don't answer the question, but have a lot of words in common with the question. This adversary does not query the model in any way." The AddOneSent examples have just one candidate sentences was picked at random. These sections were combined, added to the original SQuAD training examples and used as the new training set for fine-tuning. This added over 5,000 examples to the training dataset.

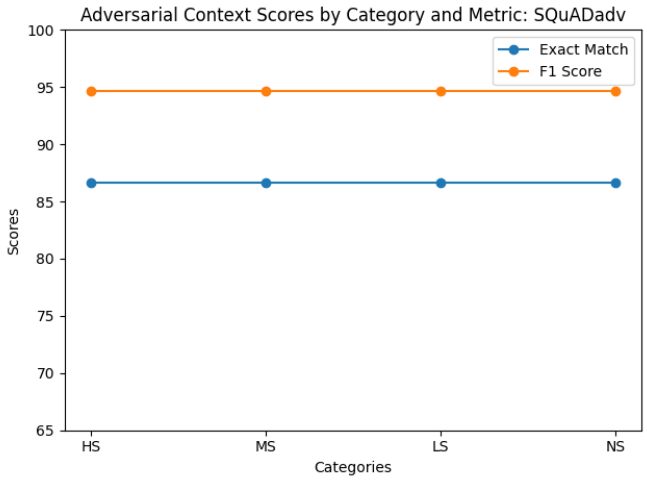
g) SQuAD V2 SQuAD Adversarial has a relatively small amount of additional examples compared to SQuAD. To address the small size of additional adversarial examples for fine-tuning, SQuAD v2 (SQuAD 2.0) was added. This version of SQuAD introduces an additional layer of complexity compared to the original SQuAD. While the SQuAD 1.0 consisted solely of questions that had corresponding answers in the associated text passages, SQuAD 2.0 added a new dimension by including questions that do not have answers within the given text. This change requires the model not only to extract correct answers but also to discern when no answer is available, a more realistic scenario. The inclusion of such unanswerable questions in SQuAD 2.0 represents a substantial challenge and a more comprehensive test of a model's natural language understanding capabilities. By fine-tuning ELECTRA on this dataset, I wanted to see if this enhanced its ability to deal with a broader range of question-answering tasks, particularly in distinguishing between answerable and unanswerable queries.

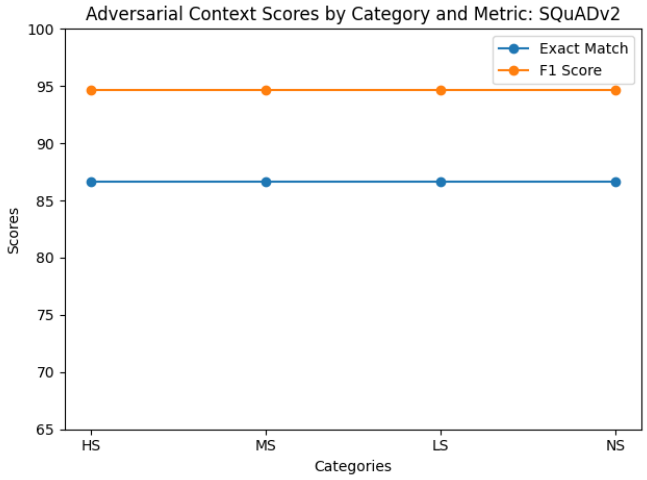
h) AdversarialQA To remedy some of the issues with data size and relevance from the previous experiments, the SQuAD/SQuAD adversarial dataset was further augmented with the adversarialQA dataset for fine-tuning. The adversarialQA dataset on Hugging Face is a collection of three distinct Reading Comprehension datasets, each constructed using an adversarial "model-in-the-loop approach." This methodology employs three different models: BiDAF, BERTLarge, and RoBERTaLarge, in the process of annotation. The dataset is divided into three parts, namely D(BiDAF), D(BERT), and D(RoBERTa), corresponding to the model used in their creation. Each of these subsets contains 10,000 training examples, along with 1,000 validation and 1,000 test examples. A key aspect of these datasets is the adversarial human annotation paradigm used in their compilation, which focuses on generating questions that are challenging for current state-of-the-art models (Bartolo et al. 2020).

The final dataset constructed for fine-tuning consisted of the SQuAD, SQuAD adversarial, and adversarialQA datasets along with the removal of SQuAD 2.0 due to the lack of quality improvements over the adversarial examples. The idea behind this combination was to enhance the signal from the adversarial questions vs that of just adding the SQuAD adversarial. This resulted in a dataset of over 128,000 examples. This combined dataset is referenced in the results as 'AdvQA'.

4 Results







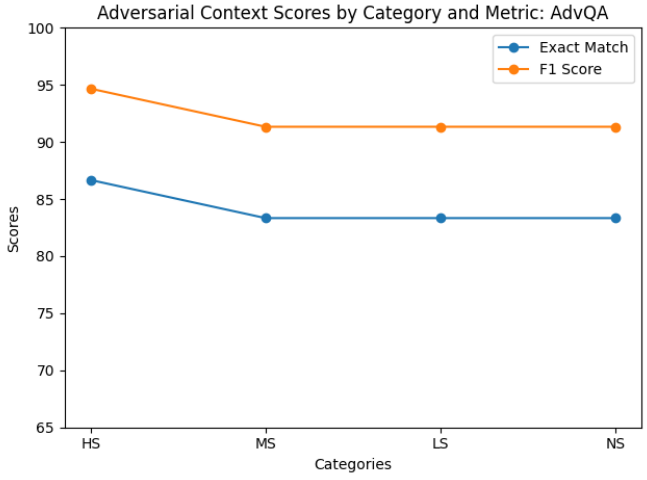


Figure 1: *Evaluation results across different adversarial context scenarios.*

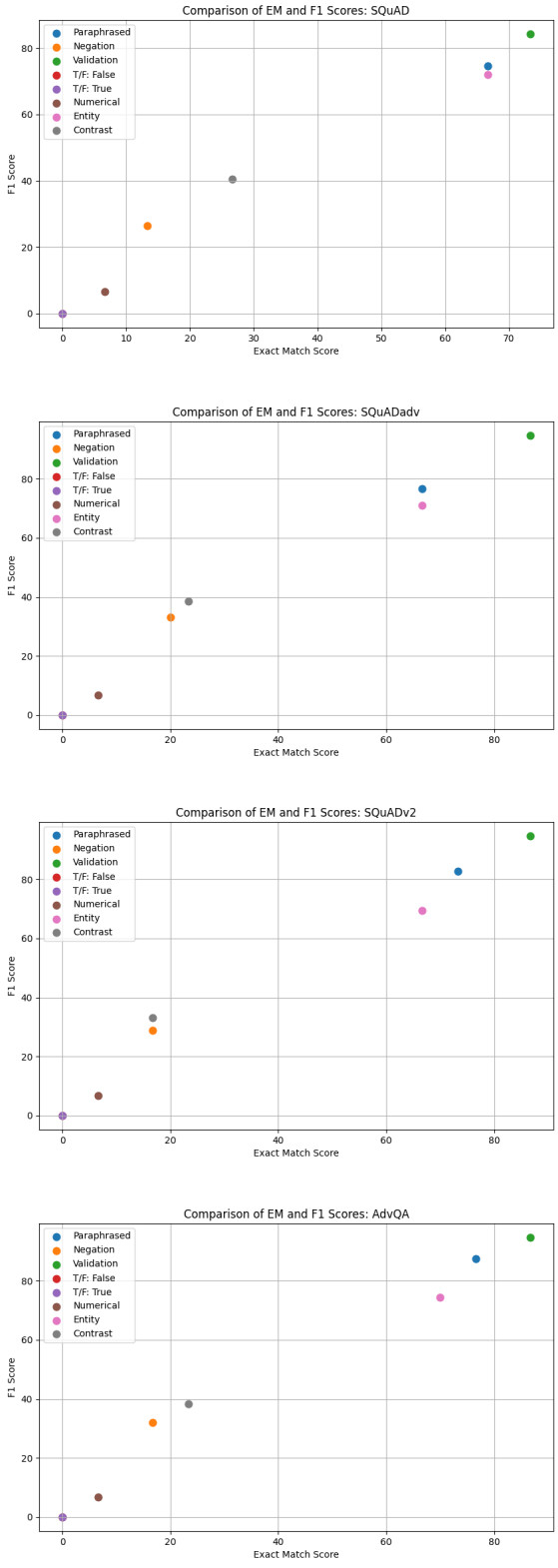


Figure 2: *F1 vs Exact Match across different Super Bowl 50 checklist sets.*

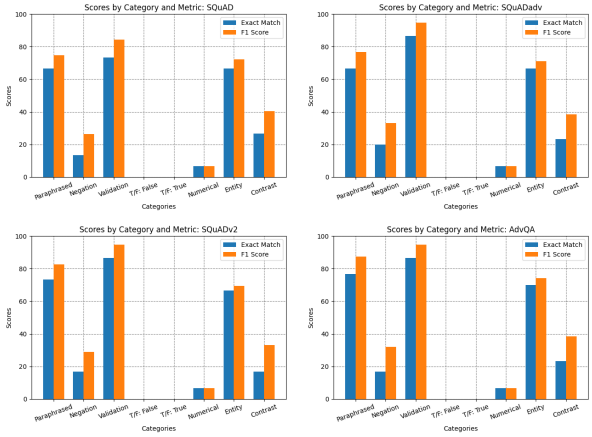


Figure 3: *F1 vs Exact Match across different Super Bowl 50 validation sets.*

Category	Scores					
	Baseline		SQuAD		SQuAD adv	
	EM	F1	EM	F1	EM	F1
Validation Set	0.0	0.0	73.33	84.26	86.67	94.67
Paraphrased	0.0	0.0	66.67	74.67	66.67	76.67
Negation	0.0	0.0	13.33	26.51	20.0	33.7
Contrast	0.0	0.0	26.67	40.56	23.33	38.56
NER	0.0	0.0	66.67	72.11	66.67	71.0
Numerical	0.0	0.0	6.67	6.67	6.67	6.67
T/F: False	0.0	0.0	0.0	0.0	0.0	0.0
T/F: True	0.0	0.0	0.0	0.0	0.0	0.0

Category	Scores			
	SQuAD v2		AdvQA	
	EM	F1	EM	F1
Validation Set	86.67	94.67	86.67	94.67
Paraphrased	73.33	82.76	76.67	87.47
Negation	16.67	28.89	16.67	32.06
Contrast	16.67	32.22	23.33	38.33
NER	66.67	69.33	70.0	74.33
Numerical	6.67	6.67	6.67	6.67
T/F: False	0.0	0.0	0.0	0.0
T/F: True	0.0	0.0	0.0	0.0

Table 1: *Exact Match and F1 Scores by Category for Different Experiments*

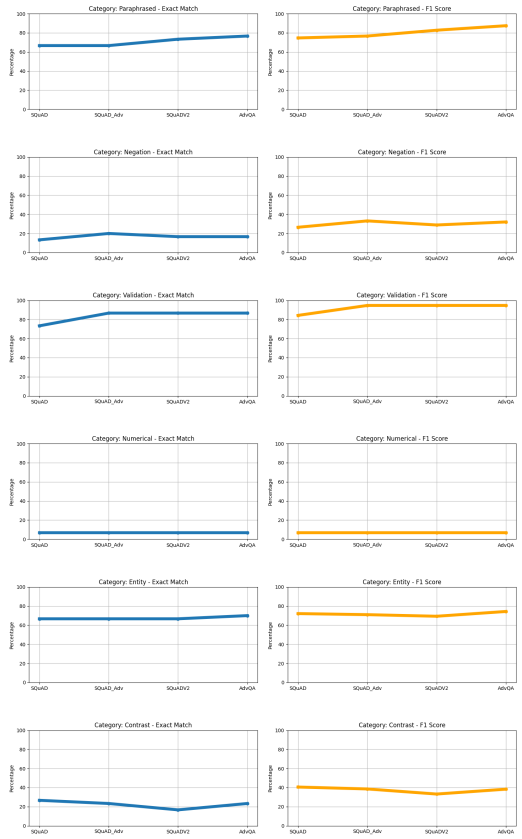


Figure 4: *F1 vs Exact Match across different Super Bowl 50 question sets.*

5 Discussion

The baseline of ELECTRA-small was truly a low bar to set for the analysis and experiments for improvement. The fine-tuned ELECTRA on the SQuAD data performed on par with other 'squad' models that have been fine tuned in the literature of around 84% model F1 score (Jia and Liang2017). Despite the great performance on the validation set, it is apparent from the different sets of altered questions that it doesn't take much variation to disrupt the QA ability of this fine tuned model. It can be seen from figure 2 and 3 that there are consistent weaknesses with the fine tuned model with regard to negation, numerical reasoning, and contrast categories. Despite these weaknesses the model did perform well with NER, contrast, and paraphrasing. This was a surprising result because it seems to have some generalization despite performing poorly in the other categories. In retrospect, the True/-False questions were not a great way to evaluate the progress of these models not only because they didn't register any correct responses but because there wasn't any True/False or Yes/No questions added to the data that helped fine tune this capability. My initial hypothesis was that the model that was fine-tuned on SQuAD v2 would have at least registered a non-zero score on one of the metrics due to the non-answering capability added by this dataset.

Incorporating SQuAD adversarial did not perform as well as initially expected but this was due to the smaller dataset size compared to the SQuAD dataset. The SQuAD with 87,000+ examples was combined with the 4,000+ adversarial examples so there really wasn't much difference between these two experiments

Across the experiments fine-tuning on the combination of SQuAD, SQuAD Adversarial, and adversarialQA produced the most improvements across the different question sets. Combining these datasets resulted in a more representative sample of the adversarial and regular training examples. The model fine-tuned on the custom AdvQA set achieved a 94.67% F1 score which exceeded the establshed human performance F1 of 91% (Jia and Liang 2017). Despite these improvements, one surprising result was that the adversarial context charts showed a decrease in scores for the adversarial scores for the different similarity examples.

6 Conclusion

Training ELECTRA-small on the combination of SQuAD and adversarialQA augmented dataset (AdvQA) showed improvements in F1 and/or Exact Match scores for NER, Paraphrased, Contrast and Negation categories. This shows that adding quality adversarial examples to the training dataset in a balanced manner helps the model generalize across multiple categories. The weaknesses exposed during the analysis on different Super Bowl 50 datasets can be remedied by fine-tuning on a more difficult dataset.

References

[1] Max Bartolo, Alastair Roberts, Johannes Welbl, Sebastian Riedel, and Pontus Stenetorp. *Beat the AI: Investigating Adversarial Human Annotation for Reading Comprehension. Trans-*

actions of the Association for Computational Linguistics, vol. 8, pp. 662-678, 2020. DOI: 10.1162/tacl_a_00338.

[2] OpenAI. *ChatGPT (Version 4.0)*. https://openai.com/, 2023.

[3] Matt Gardner, William Merrill, Jesse Dodge, Matthew E Peters, Alexis Ross, Sameer Singh, and Noah Smith. *Competency problems: On finding and removing artifacts in language data. arXiv preprint arXiv:2104.08646*, 2021.

[4] Robin Jia and Percy Liang. *Adversarial examples for evaluating reading comprehension systems*. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pp. 2021-2031, Copenhagen, Denmark, September 2017. Association for Computational Linguistics.

[5] Adam Poliak, Jason Naradowsky, Aparajita Haldar, Rachel Rudinger, and Benjamin Van Durme. *Hypothesis only baselines in natural language inference*. In *Proceedings of the Seventh Joint Conference on Lexical and Computational Semantics*, pp. 180-191, New Orleans, Louisiana, June 2018. Association for Computational Linguistics.

[6] Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. *SQuAD: 100,000+ Questions for Machine Comprehension of Text. arXiv preprint arXiv:1606.05250*, 2016.

[7] Marco Tulio Ribeiro, Tongshuang Wu, Carlos Guestrin, and Sameer Singh. *Beyond accuracy: Behavioral testing of NLP models with CheckList*. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pp. 4902-4912, Online, July 2020. Association for Computational Linguistics.

[8] Eric Wallace, Shi Feng, Nikhil Kandpal, Matt Gardner, and Sameer Singh. *Universal Adversarial Triggers for Attacking and Analyzing NLP*. Allen Institute for Artificial Intelligence; University of Maryland; Independent Researcher; University of California, Irvine. [Year of Publication].