Delving Deep into Extractive Question Answering Data

Anonymous ACL submission

Abstract

The impact of large-scale pre-trained language models on Question Answering in recent times is undeniably positive. Few prior works have attempted however to provide detailed insight into how such models learn from QA dataset component parts. For example, what specific kinds of examples are most important for models to learn from? In this paper, we examine two English QA datasets, namely SQuAD1.1 and NewsQA, and report findings on the internal characteristics of these widely employed extractive QA datasets. Experiment results reveal: (i) Models learn relatively independently of examples from outside a given question type (the performance on each question type mainly comes from that data belonging to that same question type); (ii) Increased difficulty in the training data results in better performance; (iii) Learning from QA data approximates to the process of learning question-answer matches.

1 Introduction

Large-scale pre-trained language models have come to dominate Natural Language Process research and achieve superior performance on a wide range of tasks, notably surpassing human performance with respect to several English Question Answering datasets (Devlin et al., 2019; Yang et al., 2019) such as SQuAD1.1 (Rajpurkar et al., 2016), SQuAD2.0 with unanswerable questions (Rajpurkar et al., 2018), as well as NewsQA (Trischler et al., 2017). Despite this success, as well as the large volume of research conducted on sophisticated QA systems (Zhang et al., 2020), less emphasis has been placed on effects of data used for fine-tuning and testing. A better understanding of the data has the potential to improve the generalizability of models (Rogers, 2021; Gardner et al., 2021), as well as providing helpful information for constructing datasets (Bender and Friedman, 2018; Geva et al., 2019).

Amongst NLP tasks, QA is of high interest likely due to the direct connection of QA to human comprehension. For example, several studies of QA systems and data have been carried out (Chen et al., 2016; Kaushik and Lipton, 2018), including Weissenborn et al. (2017), who reveal that employment of heuristic question type features results in competitive performance compared to sophisticated neural QA models; Jia and Liang (2017) explore the effect of adversarial examples on the performance of QA systems; Lewis et al. (2021) examine the train-test data overlap in Open Domain QA and show that QA models tend to perform much worse on examples in test data that have no overlap with the examples in training data. Furthermore, Liu et al. (2021) investigate challenging questions for QA model generalization, while Al-Negheimish et al. (2021) question the numerical reasoning ability of current QA systems by perturbing QA examples.

In this paper, we build on these earlier studies by conducting experiments with two English extractive QA datasets and QA systems and report three important findings: (i) models learn relatively independent of examples in other question types (the performance on each question type mainly comes from the data of that same question type); (ii) increased difficulty in training data improves model performance models; (iii) learning from QA data is analogous to learning question-answer matches.

2 Extractive QA Data Deep Dive

We employ QA datasets SQuAD1.1 (Rajpurkar et al., 2016) and NewsQA (Trischler et al., 2017). For SQuAD1.1 we use the official data released by Rajpurkar et al. (2016)[1] and for NewsQA we use the data from MRQA (Fisch et al., 2019)[2], and for question classification, data from Li and Roth (2002)[3], with BERT-base-uncased model

[1] https://rajpurkar.github.io/SQuAD-explorer/

[2] https://github.com/mrqa/MRQA-Shared-Task-2019

[3] https://cogcomp.seas.upenn.edu/Data/QA/QC/

		LOC	ENTY	HUM	NUM	DESC
SQuAD1.1	Train set	11.4	27.6	20.7	24.5	15.5
	Dev set	10.5	27.6	21.0	23.0	17.4
NewsQA	Train set	11.4	16.9	30.0	18.8	22.6
	Dev set	12.3	16.9	32.2	17.8	20.5

Table 1: The percentage of question types in the SQuAD1.1 and NewsQA train and dev sets.

from Huggingface (Wolf et al., 2019)[4] for both question classification and QA.

2.1 How QA models learn from different question types

QA data commonly contains a range of question types, including *when, what* and so on. This division of questions into types raises the question of *to what degree do QA systems learn from their own question type as opposed to other question types*. For example, in the case of numerical questions (how many ...?), how often, if ever, do questions of another type, such as location questions (where ...?) assist models in answering questions of this distinct type? Answering this question will help by allowing better control over the proportion of each question type employed to train QA models and improve the diversity of questions when constructing QA datasets.

We subsequently categorize questions into different classes and examine how the system learns from questions in each category. To categorise questions, we adopt question classification data (Li and Roth, 2002) to train a question classifier that categorizes questions into the following five classes: *HUM, LOC, ENTY, DESC, NUM* (Zhang and Lee, 2003),[5] and partition the QA training data into five classes before training five separate QA models for increasing data sizes from 500 to 8000, one for each question type. The dev data is also split into five classes and each QA model is applied to each subset.

Question type proportions for SQuAD1.1 and NewsQA are shown in Table 1, with a high proportion of *ENTY* and *NUM* questions in SQuAD1.1, while NewsQA has more *HUM* and *DESC* questions. A visualisation of the resulting F-1 scores of each of the five QA systems is shown in Figure 1, for both SQuAD1.1 and NewsQA for increasing amounts of training data, revealing that a QA system learns to answer a certain type of question mainly from the examples of the same question type - this is particularly true for *HUM* and *NUM* questions in SQuAD1.1 and *HUM*, *LOC* and *NUM* questions in NewsQA. Taking *NUM* questions as an example, rightmost plots in Figure 1 show that performance on distinct question types only results in a minor improvement compared to performance improvements on that question type (*NUM*).In other words, the QA system gets most of the knowledge it needs to answer *NUM* questions from the *NUM* training examples and a similar pattern is also present for other question types.

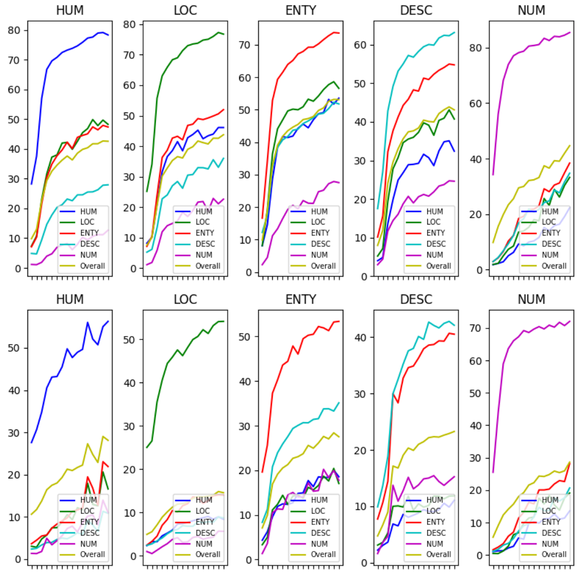

Figure 1: Visualization of F-1 learning curves for five QA systems trained on five question types (*HUM,LOC,ENTY,DESC,NUM*), tested on the dev sets for each question type and the original dev set. SQuAD1.1 (top) and NewsQA (bottom)

2.2 How a QA model learns from *difficult* and *easy* examples?

A further important aspect of QA data is the degree of the lexical overlap between the context and question in each QA example and its effects on QA system performance. We subsequently examine the effect of context in QA learning by restricting the context from which models learn.

Context-question overlap We define the QA examples with high context-question lexical overlap as *easy* examples, as increased context-question lexical overlap provides stronger clues from which a QA system can find the answer. The QA examples with low context-question lexical overlap are *difficult* examples. Inspired by Hong et al. (2020), we measure lexical overlap using BLEU

[4] https://huggingface.co/bert-base-uncased

[5] Definitions and examples provided in Appendix A.1

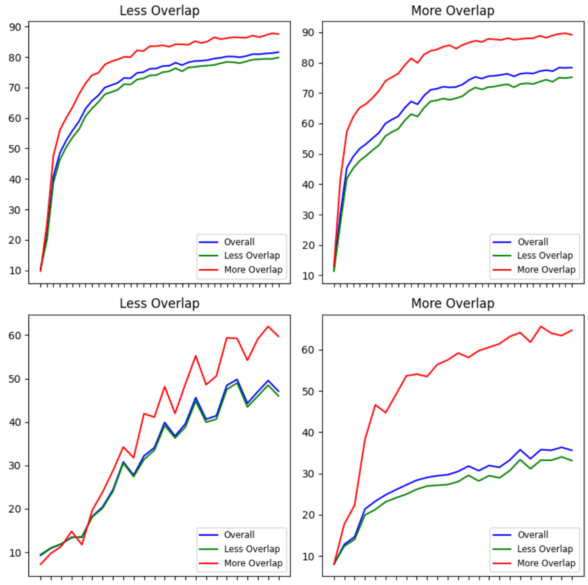

Figure 2: Visualization of F-1 score change over different lexical overlap levels and overall dev set with increased data size on *Less Overlap* and *More Overlap* SQuAD1.1 (top) and NewsQA (bottom)

scores which are calculated by BLEU-3 score using NLTK (Bird, 2006; Bird et al., 2009) [6]. Next, we divide all QA examples according to their BLEU score and train a QA model on *difficult* and *easy* examples separately.

The results on SQuAD1.1 and NewsQA are shown in Figure 2. With the same amount of data, the QA system trained on QA examples with less context-question overlap (difficult questions) across the board yields improved performance compared to the QA system trained on (easy) QA examples with more context-question overlap.

Single Sentence Context We additionally modify the *context* of QA examples to *single-sentence context* which means only keeping the sentence in the original context that contained the answer.The *single-sentence context* examples are considered *easy* examples since shorter context makes it easier to locate the correct answer whereas the *original context* examples are considered *difficult* examples. Results in Table 2 show that the performance on *single-sentence* test data is consistently better than the performance on the *original* test data.

2.3 Question-answer match

In order to investigate the degree to which models learn by memorizing question-answer matches we carry out an experiment in which we corrupt the semantics of answer text. We propose three simple strategies to perturb/corrupt answers in training QA examples: (i) *shuffle answer words*; (ii) introduce *random tokens*, i.e. randomly generate meaningless tokens to replace the original answers; (iii) *remove sentence internal words*, i.e. remove all the words in answers except the initial and final token.

		Dev-original	Dev-single-sent
SQuAD1.1	Train-original	80.61/88.25	81.75/89.50
	Train-single-sent	75.61/83.64	81.49/89.34
NewsQA	Train-original	49.55/64.53	60.51/79.18
	Train-single-sent	36.39/50.00	62.73/80.85

Table 2: Evaluation results (EM/F-1) of single-sentence context and original context QA examples on SQuAD1.1 and NewsQA.

		Overall
SQuAD1.1	Original	66.97/80.96
	Shuffle words	59.17/77.47
	Random tokens	55.99/61.40
	Remove inner words	67.61/77.78
NewsQA	Original	49.22/64.53
	Shuffle words	36.25/54.71
	Random tokens	31.72/35.91
	Remove inner words	40.29/48.20

Table 3: Evaluation results (EM/F-1) on dev sets of SQuAD1.1 and NewsQA with corrupted answers[7]

Generally speaking, an ideal QA system could be expected to be able to find the correct answers using clues from the context rather than answers alone. Corrupting answers in test QA examples therefore allows us to examine the degree to which the QA system is able to draw from the context (in other words, make use of clues from the context). Such corrupted QA examples are answerable for humans, for example below:

Context: *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* ***jysbdefziqvzbi*** *defeated the National Football Conference (NFC) champion Carolina Panthers 24–10 to earn their third Super Bowl title.*

Question: *Which NFL team won Super Bowl 50?*

Original answer: *Denver Broncos*

Corrupted/correct answer: *jysbdefziqvzbi*

Humans can easily find the correct answer - ***jysbdefziqvzbi*** even if is a meaningless word. We aim to examine whether a QA system is capable of finding

[6] https://www.nltk.org/_modules/nltk/translate/bleu_score.html

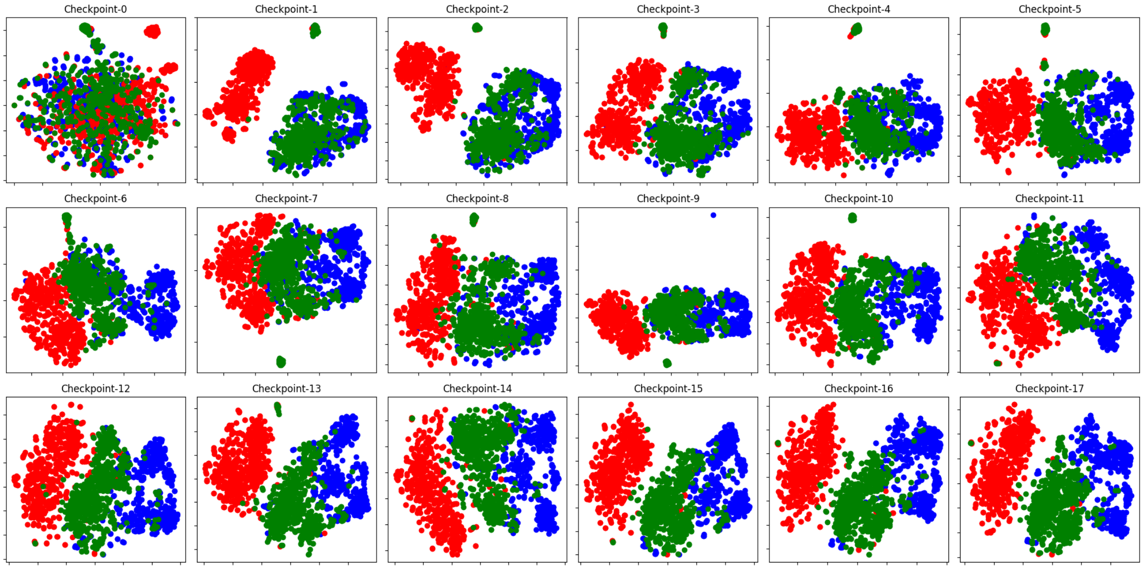

Figure 3: t-SNE visualization of randomly sampled *Answer, Question* and *Context* wordpiece representations from 18 checkpoints in training process starting from checkpoint-0 (the vanilla BERT) to checkpoint-17 (the BERT finetuned on SQuAD1.1), where *Answer* is in *blue*, *Question* is in *red*, *Context* is in *green*.

such corrupted correct answers.

The average results of three runs on SQuAD1.1 and NewsQA are shown in Table 3. We find that corrupting the semantic information of answer text (especially *random tokens*) results in a substantial performance drop (~25% drop for SQuAD1.1 and ~50% drop for NewsQA) – the margin is larger for NewsQA (~30 F-1 score drop). Moreover, to further gain insight into the representations learned by the QA system, we randomly sampled 500 context-question-answer Wordpiece (Wu et al., 2016) representations from 18 checkpoints of BERT model during the fine-tuning process on SQuAD1.1 [8] and use t-SNE (van der Maaten and Hinton, 2008) to visualize them – see Figure 3. The visualization clearly shows the learning process of the QA system: (i) the representations of questions (red) are differentiated from the representations of context (green) and answers (blue), and this is probably the result of the different segment vector added to question and context (answer is in context); (ii) as the fine-tuning process continues, the representations of context and answer are gradually separated.

3 Discussion and Conclusion

We presented a series of experiments investigating the internal characteristics of two popular extractive QA datasets: SQuAD1.1 and NewsQA. The question type experiments show that models learn relatively independent of examples in other question types, especially for QA examples in *HUM, LOC, ENTY, NUM* - and the effect is more extreme for NewsQA. Furthermore, we found that the models trained on *difficult* QA examples (low context-question lexical overlap and longer context) yield better performance compared to those trained on *easy* QA examples. These two findings reveal how models learn from QA examples of different question types as well as different difficulty levels, providing useful information on how to promote question diversity and reduce context-question overlap when constructing QA datasets. Finally, the results of the question-answer match experiments show that answer perturbation causes substantial performance drop, demonstrating that models heavily rely on the clues from answer text rather than the clues from context. This suggests the need to build QA models with more comprehension rather than simply memorizes question-answer matches. In future work, we aim to apply our analysis to multilingual data to explore how QA models behave across different languages and we plan to investigate more diverse QA data beyond extractive QA data.

References

Hadeel Al-Negheimish, Pranava Madhyastha, and Alessandra Russo. 2021. Numerical reasoning in ma-

[8] checkpoints of 0, 100, 200, 300, 400, 500, 600, 700, 800, 900, 1000, 2000, 3000, 4000, 5000, 6000, 7000 step during the training process and the finetuned model

chine reading comprehension tasks: are we there yet? In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 9643–9649, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Emily M. Bender and Batya Friedman. 2018. Data statements for natural language processing: Toward mitigating system bias and enabling better science. *Transactions of the Association for Computational Linguistics*, 6:587–604.

Steven Bird. 2006. Nltk: the natural language toolkit. In *Proceedings of the COLING/ACL 2006 Interactive Presentation Sessions*, pages 69–72.

Steven Bird, Ewan Klein, and Edward Loper. 2009. *Natural language processing with Python: analyzing text with the natural language toolkit*. " O'Reilly Media, Inc.".

Danqi Chen, Jason Bolton, and Christopher D. Manning. 2016. A thorough examination of the CNN/Daily Mail reading comprehension task. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2358–2367, Berlin, Germany. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Adam Fisch, Alon Talmor, Robin Jia, Minjoon Seo, Eunsol Choi, and Danqi Chen. 2019. MRQA 2019 shared task: Evaluating generalization in reading comprehension. In *Proceedings of 2nd Machine Reading for Reading Comprehension (MRQA) Workshop at EMNLP*.

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

Mor Geva, Yoav Goldberg, and Jonathan Berant. 2019. Are we modeling the task or the annotator? an investigation of annotator bias in natural language understanding datasets. 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)*, pages 1161–1166, Hong Kong, China. Association for Computational Linguistics.

Giwon Hong, Junmo Kang, Doyeon Lim, and Sung-Hyon Myaeng. 2020. Handling anomalies of synthetic questions in unsupervised question answering. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 3441–3448, Barcelona, Spain (Online). International Committee on Computational Linguistics.

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

Divyansh Kaushik and Zachary C. Lipton. 2018. How much reading does reading comprehension require? a critical investigation of popular benchmarks. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 5010–5015, Brussels, Belgium. Association for Computational Linguistics.

Patrick Lewis, Pontus Stenetorp, and Sebastian Riedel. 2021. Question and answer test-train overlap in open-domain question answering datasets. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 1000–1008, Online. Association for Computational Linguistics.

Xin Li and Dan Roth. 2002. Learning question classifiers. In *COLING 2002: The 19th International Conference on Computational Linguistics*.

Linqing Liu, Patrick Lewis, Sebastian Riedel, and Pontus Stenetorp. 2021. Challenges in generalization in open domain question answering.

Pranav Rajpurkar, Robin Jia, and Percy Liang. 2018. Know what you don't know: Unanswerable questions for SQuAD. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 784–789, Melbourne, Australia. Association for Computational Linguistics.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. SQuAD: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2383–2392, Austin, Texas. Association for Computational Linguistics.

Anna Rogers. 2021. Changing the world by changing the data. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 2182–2194, Online. Association for Computational Linguistics.

Adam Trischler, Tong Wang, Xingdi Yuan, Justin Harris, Alessandro Sordoni, Philip Bachman, and Kaheer Suleman. 2017. NewsQA: A machine comprehension dataset. In *Proceedings of the 2nd Workshop on Representation Learning for NLP*, pages 191–200, Vancouver, Canada. Association for Computational Linguistics.

Laurens van der Maaten and Geoffrey Hinton. 2008. Visualizing data using t-sne. *Journal of Machine Learning Research*, 9(86):2579–2605.

Dirk Weissenborn, Georg Wiese, and Laura Seiffe. 2017. Making neural QA as simple as possible but not simpler. In *Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017)*, pages 271–280, Vancouver, Canada. Association for Computational Linguistics.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, R'emi Louf, Morgan Funtowicz, and Jamie Brew. 2019. Huggingface's transformers: State-of-the-art natural language processing. *ArXiv*, abs/1910.03771.

Yonghui Wu, Mike Schuster, Zhifeng Chen, Quoc V Le, Mohammad Norouzi, Wolfgang Macherey, Maxim Krikun, Yuan Cao, Qin Gao, Klaus Macherey, et al. 2016. Google's neural machine translation system: Bridging the gap between human and machine translation. *arXiv preprint arXiv:1609.08144*.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R Salakhutdinov, and Quoc V Le. 2019. Xlnet: Generalized autoregressive pretraining for language understanding. *Advances in neural information processing systems*, 32.

Dell Zhang and Wee Sun Lee. 2003. Question classification using support vector machines. In *Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval*, pages 26–32.

Zhuosheng Zhang, Hai Zhao, and Rui Wang. 2020. Machine reading comprehension: The role of contextualized language models and beyond.

A Appendix

A.1 Experimental Setup

A.1.1 Hyperparameters

We used `DistributedDataParallel` in `torch.nn.parallel` to run all the training scripts for question classification and QA.

Question classification model training We use `bert-based-uncased` as our question classifier, the learning rate is set to 2e-5, the maximum sequence length is set to 128, we run the training script for 3 epochs, the training was conducted on 2 GeForce GTX 1080Ti GPUs, the training batch size is 32 for each GPU.

QA model training We use `bert-based-uncased` as our QA model, the learning rate is set to 3e-5, the maximum sequence length is set to 384, the doc stride length is set to 128, we run the training script for 2 epochs for training each QA system in experiments, the training was conducted on 2 GeForce GTX 1080Ti GPUs, the training batch size is 12 for each GPU.

A.2 Question type definition and examples

We show the definitions of question type *HUM, LOC, ENTY, DESC, NUM* and some examples from the question classification data (Li and Roth, 2002) and predictions of SQuAD1.1 and NewsQA in Table 4.

A.3 QA examples with corrupted answers

We give some QA examples with corrupted answers and corresponding predicted answers in Table 5.

Question type	Definition	Examples
HUM	people, individual, group, title	What contemptible scoundrel stole the cork from my lunch ? Which professor sent the first wireless message in the USA ? Who was sentenced to death in February ?
LOC	location, city, country, mountain, state	Where is the Kalahari desert ? Where is the theology library at Notre Dame ? Where was Cretan when he heard screams ?
ENTY	animal, body, color, creation, currency, disease/medical, event, food, instrument, language, plant, product, religion, sport, symbol, technique, term, vehicle	What relative of the racoon is sometimes known as the cat-bear ? What is the world's oldest monographic music competition ? What was the name of the film about Jack Kevorkian ?
DESC	definition, description, manner, reason	What is Eagle 's syndrome styloid process ? How did Beyonce describe herself as a feminist ? What are suspects blamed for ?
NUM	code, count, date, distance, money, order, other, percent, period, speed, temperature, size, weight	How many calories are there in a Big Mac ? What year did Nintendo announce a new Legend of Zelda was in the works for Gamecube ? How many tons of cereal did Kelloggs donate ?

Table 4: Definition of each question type and corresponding examples in SQuAD1.1 and NewsQA.

Context	Question	Original Answer	Corrupted Answer	Predicted Answer
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 jysbdefziqvzbi 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.	Which NFL team won Super Bowl 50?	Denver Broncos	jysbdefziqvzbi	Carolina Panthers
Luther is honoured on hzhttbntedf with a commemoration in the Lutheran Calendar of Saints and in the Episcopal (United States) Calendar of Saints. In the Church of England's Calendar of Saints he is commemorated on 31 October.	When is Luther commemorated in the Lutheran Calendar of Saints ?	18 February	hzhttbntedf	31 October
Many famous potters, such as Josiah Wedgwood, William De Morgan and Bernard Leach as well as Mintons & Royal Doulton are represented in the collection. There is an extensive collection of Delftware produced in both Britain and Holland, which includes a circa 1695 flower pyramid over a metre in height. Bernard Palissy has several examples of his work in the collection including dishes, jugs and candlesticks. The largest objects in the collection are a series of elaborately ornamented ceramic stoves from the 16th and 17th centuries, made in yizzzqmwoibvwvdnvx-soalb. There is an unrivalled collection of Italian maiolica and lustreware from Spain. The collection of Iznik pottery from Turkey is the largest in the world.	The largest objects in the V&A ceramics and glass collection were produced in which countries?	Germany and Switzerland	yizzzqmwoi bvwvdnvx-soalb	Britain and Holland

Table 5: Some QA examples with corrupted answers and corresponding predicted answers