DISFL-QA: A Benchmark Dataset for Understanding Disfluencies in Question Answering

Aditya Gupta[♠] **Jiacheng Xu**[◇*] **Shyam Upadhyay**[♠] **Diyi Yang**[♣] **Manaal Faruqui**[♠]
[♠]Google Assistant
[◇]The University of Texas at Austin
[♣]Georgia Institute of Technology
disfl-qa@google.com

Abstract

Disfluencies is an under-studied topic in NLP, even though it is ubiquitous in human conversation. This is largely due to the lack of datasets containing disfluencies. In this paper, we present a new challenge question answering dataset, DISFL-QA, a derivative of SQUAD, where humans introduce contextual disfluencies in previously fluent questions. DISFL-QA contains a variety of challenging disfluencies that require a more comprehensive understanding of the text than what was necessary in prior datasets. Experiments show that the performance of existing state-of-the-art question answering models degrades significantly when tested on DISFL-QA in a zero-shot setting. We show data augmentation methods partially recover the loss in performance and also demonstrate the efficacy of using gold data for fine-tuning. We argue that we need large-scale disfluency datasets in order for NLP models to be robust to them. The dataset is publicly available at: https://github.com/google-research-datasets/disfl-qa.

1 Introduction

During conversations, humans do not always premeditate exactly what they are going to say; thus a natural conversation often includes interruptions like repetitions, restarts, or corrections. Together these phenomena are referred to as *disfluencies* (Shriberg, 1994). Figure 1a shows different types of conventional disfluencies in an utterance, as described by Shriberg (1994).

With the growing popularity of voice assistants, such disfluencies are of particular interest for goal-oriented or information seeking dialogue agents, because an NLU system, trained on fluent data, can easily get misled due to their presence. Figure 1b shows how the presence of *disfluencies* in a question answering (QA) setting, namely SQUAD (Rajpurkar et al., 2018), affects the prediction of a state-of-the-art T5 model (Raffel et al., 2020). For example, the original question q_1 is seeking an answer about the location of *Normandy*. In the disfluent version dq_1 (which is **semantically equivalent** to q_1), the user starts asking about *Norse* and then corrects themselves to ask about the *Normandy* instead. The presence of this correctional disfluency confuses the QA model, which tend to rely on shallow textual cues from question for making predictions.

Repetition	When is **Eas ugh Easter** this year?
Correction	When is **Lent I meant Easter** this year?
Restarts	**How much no wait when is** Easter this year?

(a) Conventional categories of *Disfluencies*. The ***reparandum*** (words intended to be corrected or ignored), ***interregnum*** (optional discourse cues) and ***repair*** are marked.

Passage: *The Normans (Norman: Nourmands; French: Normands; Latin: Normanni) were the people who in the 10th and 11th centuries gave their name to Normandy, a region in France. They were descended from Norse ("Norman" comes from "Norseman") raiders and pirates from Denmark, Iceland and Norway who, under their leader Rollo, ...*

$\mathbf{q_1}$: In what country is Normandy located?
$\mathbf{dq_1}$: In what country is **Norse** found **no wait Normandy not Norse**?
$\mathbf{T5(q_1)}$: France ✓
$\mathbf{T5(dq_1)}$: Denmark ✗

$\mathbf{q_2}$: When were the Normans in Normandy?
$\mathbf{dq_2}$: **From which countries no tell me when were the Normans in Normandy**?
$\mathbf{T5(q_2)}$: 10th and 11th centuries ✓
$\mathbf{T5(dq_2)}$: Denmark, Iceland and Norway ✗

(b) Contextualized *Disfluencies* in DISFL-QA (§2).

Figure 1: **(a)** Categories of disfluencies (Shriberg, 1994) **(b)** A passage and questions (q_i) from SQUAD, along with their disfluent versions (dq_i) and predictions from a T5-QA model.

[*]Work done during an internship at Google.

Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021, pages 3309–3319

Unfortunately, research in NLP and speech community has been impeded by the lack of curated datasets containing such disfluencies. The datasets available today are mostly conversational in nature, and span a limited number of very specific domains (e.g., telephone conversations, court proceedings) (Godfrey et al., 1992; Zayats et al., 2014). Furthermore, only a small fraction of the utterances in these datasets contain disfluencies, with a limited and skewed distribution of disfluencies types. In the most popular dataset in the literature, the SWITCHBOARD corpus (Godfrey et al., 1992), only 5.9% of the words are *disfluencies* (Charniak and Johnson, 2001), of which > 50% are *repetitions* (Shriberg, 1996), which has been shown to be the relatively simpler form of disfluencies (Zayats et al., 2014; Jamshid Lou et al., 2018; Zayats et al., 2019).

To fill this gap, we present DISFL-QA, the first dataset containing *contextual disfluencies* in an information seeking setting, namely question answering over Wikipedia passages. DISFL-QA is constructed by asking human raters to insert disfluencies in *questions* from SQUAD-v2, a popular question answering dataset, using the passage and remaining questions as context. These contextual disfluencies lend naturalness to DISFL-QA, and challenge models relying on shallow matching between question and context to predict an answer. Some key properties of DISFL-QA are:

- DISFL-QA is a targeted dataset for disfluencies, in which all questions (≈12k) contain disfluencies, making for a much larger disfluent test set than prior datasets.
- Over 90% of the disfluencies in DISFL-QA are corrections or restarts, making it a much harder test set for disfluency correction (§2.2).
- DISFL-QA contains wider diversity in terms of semantic distractors than earlier disfluency datasets, and newer phenomenon such as coreference between the *reparandum* and the *repair* (§2.3).

We experimentally reveal the brittleness of state-of-the-art LM based QA models when tested on DISFL-QA in zero-shot setting (§4.1). Since collecting large supervision datasets containing disfluencies for training is expensive, different data augmentation methods for recovering the zero-shot performance drop are also evaluated (§3.3). Finally, we demonstrate the efficacy of using the human annotated data in varying fractions, for both end-to-end QA supervision and disfluency generation based data augmentation techniques (§4.2).

We argue that creation of datasets, such as DISFL-QA, are vital for **(1)** improving understanding of disfluencies, and **(2)** developing robust NLU models in general.

2 DISFL-QA: Adding Disfluencies to QA

DISFL-QA builds upon the existing SQUAD-v2 dataset, a question answering dataset which contains curated paragraphs from Wikipedia and associated questions. Each question associated with the paragraph is sent for a human annotation task to add a contextual disfluency using the paragraph as a source of distractors. Finally, to ensure the quality of the dataset, a subsequent round of human evaluation with an option to re-annotate is conducted.

2.1 Source of Questions

We sourced passages and questions from SQUAD-v2 (Rajpurkar et al., 2018) development set. SQUAD-v2 is an extension of SQUAD-v1 (Rajpurkar et al., 2016) that contains unanswerable questions written adversarially by crowd workers to look similar to answerable ones from SQUAD-v1. We use both answerable and unanswerable questions for each passage in the annotation task.

2.2 Annotation Task

To ensure high quality of the dataset, our annotation process consists of 2 rounds of annotation:

First Round of Annotation. Expert raters were shown the passage along with all the associated questions and their answers, with one of the question-answer pair highlighted for annotation.[1] The raters were instructed to use the provided context in crafting disfluencies to make for a non-trivial dataset.

The rater had to provide a disfluent version of the question that (a) is *semantically equivalent* to the original question (b) is *natural*, i.e., a human can utter them in a dialogue setting. When

[1]The raters were linguistic experts, and were trained for the task with 2 rounds of pilot annotation.

Type	Passage (some parts shortened)	Fluent Question	Disfluent Question
Interrogative Restart (**30%**)	*. . . Roger de Tosny travelled to the Iberian Peninsula to carve out a state for himself. In 1064, during the War of Barbastro, William of Montreuil led the papal army . . .*	Who was in charge of the papal army in the War of Barbastro?	**Where did the** no who was in charge of the papal army in the Barbastro War?
Entity Correction (**25.6%**)	*. . . While many commute to L.A. and Orange Counties, there are some differences in development, as most of San Bernardino and Riverside Counties were developed in the 1980s and 1990s. . .*	Other than the 1980s, in which decade did most of San Bernardino and Riverside Counties develop?	Other than **the 1990s I mean actually the 1980s** which decade did San Bernardino and Riverside counties develop?
Adverb/Adj. Correction (**20%**)	*. . . Southern California is home to Los Angeles International Airport, the second-busiest airport in the United States by passenger volume; San Diego International Airport the busiest single runway airport in the world. . .*	What is the second busiest airport in the United States?	What airport in the United States is the **busiest no second busiest**?
Entity Type Correction (**21.1%**)	*. . . To the east is the Colorado Desert and the Colorado River, and the Mojave Desert at the border with Nevada. To the south is the Mexico-United States border. . .*	What is the name of the water body that is found to the east?	What is the name of the **desert wait the water body** that is found to the east?
Others (**3.3%**)	*. . . Complexity measures are very generally defined by the Blum complexity axioms. Other complexity measures used in complexity theory include communication complexity and decision tree complexity. . .*	What is typically used to broadly define complexity measures?	What is **defined no is typically used** to broadly define complexity measures?

Table 1: Example passage and fluent questions from the SQuAD dataset and their disfluent versions provided by human raters, categorized by the type of disfluency along with their estimated percentage in the DISFL-QA dataset.

writing the disfluent version of a question, we instructed raters not to include partial words or filled pauses (e.g., "*um*", "*uh*", "*ah*" etc.), as they can be detected relatively easily (Johnson and Charniak, 2004; Jamshid Lou and Johnson, 2017). Raters were shown example disfluencies from each of the categories in Table 1. On average, raters spent 2.5 minutes per question. Introduction of a disfluency increased the mean length of a question from 10.3 to 14.6 words.

Human Evaluation + Re-annotation. To assess and ensure high quality of the dataset, we asked a another set of human raters the following yes/no questions:

1. Is the disfluent question *consistent* with respect to the fluent question? i.e., the disfluent question is semantically equivalent to the original question in that they share the same answer.

2. Is the disfluent question *natural*? Naturalness is defined in terms of human usage, grammatical errors, meaningful distractors etc.

After the first round of annotation, we found that the second pool of raters found the disfluent questions to be consistent and natural 96.0% and 88.5% of the time, with an inter-annotator agreement of 97.0% and 93.0%[2], respectively. This suggests that the initial round of annotation resulted in a high quality dataset. Furthermore, for the cases identified as either inconsistent or unnatural, we conducted a second round of re-annotation with updated guidelines to make required corrections.

2.3 Categories of Disfluencies

To assess the distribution of different types of disfluencies, we sampled 500 questions from the training and development sets and manually annotated the nature of disfluency introduced by the raters. Table 1 shows the distribution of these categories in the dataset.

A notable difference between DISFL-QA and SWITCHBOARD (Godfrey et al., 1992) is that DISFL-QA contains a larger fraction of corrections and restarts, which have been shown to be the hardest disfluencies to detect and correct (Zayats et al., 2014; Jamshid Lou et al., 2018; Yang et al., 2020). From Table 1, we can see that ≈30% and >65% of the disfluencies in DISFL-QA are restarts and corrections respectively.

In addition to the specific categories men-

[2]Cohen's κ = 0.55, indicating moderate agreement.

Dataset	Switchboard	DISFL-QA
Domain	Telephonic Conversations	Wikipedia Passages
Goal-oriented	No	Yes
Contextual	No	Yes
Size (# sentences)	7.9k	11.8k
Disfluencies	20%	100%
Correction & Restarts	<50%	>90%
Coreferences	<1%	≈10%

Table 2: Comparison of DISFL-QA with SWITCHBOARD. DISFL-QA is more diverse, contains harder disfluencies and new phenomenon like coreference.

Rule	Fluent	Disfluent
Q	What was the Norman religion?	What was **replaced with** no no what was the Norman religion?
V	When was the Duchy of Normandy founded?	When was the Duchy of Normandy **offered** ugh I mean **founded**?
ADJ	What is the original meaning of the word Norman?	What is the **English** rather **original** meaning of the word Norman?
ADV	Who did Beyoncé perform privately for in 2011?	Who did Beyoncé perform **publicly** oops **privately** for in 2011?
ENT	Who was a prominent Huguenot in Holland?	Who was a prominent **Saint Nicholas** no I mean Huguenot in Holland?

Table 3: Example of synthetically generated disfluent questions using the contextual heuristics.

tioned in Table 1, the dataset includes other challenging phenomena which are shared across these categories. For instance, example below shows disfluencies which introduce *coreferences* between the *reparandum* and the *repair* (mentions marked **[.]**), allowing more complex corrections not present in existing datasets:

Who does BSkyB have an operating license from ? → **Who removed [BSkyB's] operating license** no scratch **that who do [they] have [their] operating license from** ?

Table 2 summarizes the key differences between DISFL-QA and the SWITCHBOARD dataset.

3 Experimental Setup

3.1 Models to Compare

We use two different modeling approaches to answer disfluent questions in DISFL-QA.

LMs for QA. We use BERT (Devlin et al., 2019) and T5 (Raffel et al., 2020) as our QA models in the standard setup which has shown to achieve state-of-the-art performance for SQUAD. We fine-tune BERT for a span selection task, whereby predicting `start` and `end` probabilities for all the tokens in the context.

T5 is finetuned under the standard *text2text* formulation, when given (question, passage) as input the model generates the answer as the output. For predicting `<no answer>`, the model was trained to generate *"unknown"*.

LMs for Disfluency Correction. We also finetune the above LMs as disfluency correction models. Given the disfluent question as input, a correction model predicts the fluent question, which is then fed into a QA model. For BERT, we use the state-of-the-art BERT-based disfluency correction model by Jamshid Lou and Johnson (2020) trained on SWITCHBOARD. We also train T5 models on DISFL-QA to prevent the distribution skew between SWITCHBOARD and DISFL-QA, and account for new phenomena like coreferences.

3.2 Training Settings

We train the BERT and T5 variants on the following two data configurations:

ALL where the model is trained on all of SQUAD-v2, including the non-answerable questions. Evaluation is done against the entire test set.

ANS where the model is trained only on answerable questions from SQUAD-v1, without the capabilities of handling non-answerable questions.

3.3 Datasets

Human Annotated Datasets. We use 3 datasets in our experiments: SQUAD-v1, SQUAD-v2, and DISFL-QA. We split the $11,825$ annotated questions in DISFL-QA into train/dev/test set containing $7182/1000/3643$ questions, respectively. The split was also done at an article level such that the questions belonging to the same passage belong in the same split. For zero-shot experiments, we only use the train of SQUAD.

Evaluation is done on the subset of SQuAD-v2 development set that corresponds to the DISFL-QA test to ensure fair comparison.

Heuristically Generated Data. We also generate disfluencies heuristically to validate the importance of human annotated disfluencies. Inspired by the disfluency categories seen in our annotation task, we derive the following heuristics to

Model	Train	Eval	HasAns-F1	NoAns-F1	Overall-F1
BERT-QA	**ALL**	SQuAD	83.87	70.55	77.46
		Heuristics	51.45 ↓ 32.42	74.49 ↑ 3.94	62.53 ↓ 14.93
		DISFL-QA	40.97 ↓ 42.90	75.97 ↑ 5.42	57.81 ↓ 19.65
	ANS	SQuAD	89.63	-	89.63
		Heuristics	80.52 ↓ 9.11	-	80.52 ↓ 9.11
		DISFL-QA	78.88 ↓ 10.75	-	78.88 ↓ 10.75
T5-QA	**ALL**	SQuAD	91.38	87.67	89.59
		Heuristics	39.98 ↓ 51.40	92.57 ↑ 4.90	65.27 ↓ 24.32
		DISFL-QA	35.31 ↓ 56.07	90.06 ↑ 2.39	61.64 ↓ 27.95
	ANS	SQuAD	93.71	-	93.71
		Heuristics	81.73 ↓ 12.01	-	81.73 ↓ 12.01
		DISFL-QA	80.39 ↓ 13.32	-	80.39 ↓ 13.32
Disfluency Correction + T5-QA	**ALL**	SQuAD	91.38	87.67	89.59
		Heuristics	42.83 ↓ 48.55	92.18 ↑ 4.51	66.56 ↓ 23.03
		DISFL-QA	43.61 ↓ 47.77	89.55 ↑ 1.88	65.71 ↓ 23.88
	ANS	SQuAD	93.71	-	93.71
		Heuristics	82.27 ↓ 10.44	-	82.27 ↓ 10.44
		DISFL-QA	82.64 ↓ 11.07	-	82.64 ↓ 11.07

Table 4: Breakdown of zero-shot performance of fine-tuned BERT and T5 QA models, trained only on the SQuAD dataset, and evaluated on SQuAD, Heuristics (§3.3), and DISFL-QA test sets. We also evaluate the performance by using state-of-the-art disfluency detection model by Jamshid Lou and Johnson (2020) in a pipelined fashion.

augment our data with *silver*[3] standard disfluencies: (i) SWITCH-Q which inserts prefix of another question as a prefix to the original question, and (ii) SWITCH-X, where X could be verb, adjective, adverb, or entity, and is inserted as a reparandum in the question.

To facilitate contextual disfluencies, we use the reparandums from the context. For SWITCH-VERB/ADJ/ADV/ENT, this was done by picking tokens and phrases from the context passage. For SHIFT-Q, we used other questions associated with the same passage. We used spaCy[4] NER and POS tagger to extract relevant entities and POS tags, and sample interregnum from a list of fillers. Table 3 shows an example from each of the heuristics. We then finally combine all the heuristics (ALL in Table 3) by uniformly sampling a single disfluent question from the set of possible transformations of the question.

3.4 Evaluation Method

In all our experiments, we evaluate QA performance using the standard SQuAD-v2 evaluation script which reports EM and F1 scores over the `HasAns` (asnwerable) and `NoAns` (non-answerable) slices along with the overall scores. For brevity, we report only the F1 numbers as we observed similar trends in EM and F1 across our experiments.

[3]The *silver* nature of the data is due to the fact that we can not enforce naturalness or semantic equivalence of §2.

[4]https://spacy.io/

4 Experiments

We conduct experiments with DISFL-QA to answer the following questions: **(a)** Are state-of-the-art LM based QA models robust to introduction of disfluencies in the questions under a zero-shot setting ? **(b)** Can we use heuristically generated synthetic disfluencies to aid the training of QA models to handle disfluencies ? **(c)** Given a small amount of labeled data, can we recover performance by fine-tuning the QA models or training a disfluency correction model to pre-process the disfluent questions into fluent ones before inputting to the QA models ? **(d)** In the above setting, can we train a generative model to generate more disfluent training data ?

4.1 Zero-Shot Performance

Table 4 shows the performance of different variants measuring their zero-shot capabilities.

Performance of BERT-QA and T5-QA. We see from Table 4 that when tested directly on on heuristics and DISFL-QA test sets, both the BERT-QA and T5-QA models exhibit significant performance drop, as compared to the performance on the fluent benchmark of SQuAD. The performance drop for the complete models is greater

Original	HasAns		NoAns
Prediction	NoAns	WrongAns	HasAns
SQuAD	71	150	216
DISFL-QA	1091	168	174

Table 5: Breakdown of prediction errors for the T5-QA-ALL model on the fluent and disfluent questions. WrongAns represents that the model predicted an incorrect span from context.

when compared to their answerable-only counterparts. The best performing T5-ALL model shows a **drop of 27.95 F1** points for the complete setup and **13.32 F1** point for the answerable only T5-ANS model. This shows BERT and T5 are not robust when questions contain disfluencies.

Disfluency Correction + T5-QA. We use the BERT based state-of-the-art disfluency correction (Jamshid Lou and Johnson, 2020) as a preprocessing step before feeding the input to our T5-QA model. The models trained on SWITCHBOARD are not able to fill a significant performance gap, with the complete and answerable models recovering 4.07 and 2.25 F1 points, respectively. We will revisit this setting in the few-shot experiments.

DISFL-QA test-set vs. Heuristics test-set. Next, we compare the performance of heuristically generated disfluent questions against the human annotated questions. In general, human annotated disfluent questions exhibit larger performance drop compared to heuristics, across different models.

Taking a closer look at the T5-ALL model shows that DISFL-QA shows a bigger drop in HasAns cases and smaller increase in NoAns cases, as compared to the heuristics test set. For the T5-ANS model, DISFL-QA shows a larger drop in performance which is attributed to the model picking wrong answer span. Based on this, we hypothesize that between the two datasets, heuristics are able to confuse the models in over-predicting <no answer>, but DISFL-QA is superior when it comes to confuse the models to picking a different answer span altogether (as seen in Table 4 for models in ANS setting). This demonstrates that collecting a dataset like DISFL-QA via human annotation holds value for contextual disfluencies.

	HasAns F1	NoAns F1	Overall F1
Fluent (⋆)	91.38	87.67	89.59
Zero-Shot	35.21	90.06	61.64
+ SW-ADJ	68.49	86.24	77.03
+ SW-ADV	67.37	85.27	75.98
+ SW-ENT	**74.76**	**85.95**	**80.14**
+ SW-Q	70.03	78.94	74.31
+ SW-VERB	68.01	87.16	77.22
+ ALL	**78.86**	**85.96**	**82.27**

Table 6: Performance on DISFL-QA with individual (SW-XX) and combined (ALL) heuristics based data augmentation and fine-tuning.

Performance Gap Breakdown. For models trained on ALL setting, we find that the performance drop is largely due to the drop in F1 (over **50** points) on HasAns questions as opposed to NoAns questions, where it is almost negligible or even positive in some cases. Upon closer analysis (Table 5) we find that a major fraction of prediction errors for HasAns is attributed to HasAns → NoAns errors, instead of HasAns → WrongAns.[5]

We believe that the disfluencies are causing the answerable questions to resemble the non-answerable ones as seen by both BERT and T5 models under ALL setting. This results in an overly conservative model in terms of answerability and instead resorts to over-predicting <no answer>, causing gain in non-answerable recall at the cost of precision. In contrast, for a comparable ANS model the drop in F1 is smaller, primarily due to relatively easier decision making, i.e. not required to decide when to answer vs. not.

Fine-tuning on Heuristic Data. In this experiment, we fine-tune on heuristically generated data from §3.3 and directly test on DISFL-QA. Table 6 compares the performance of the heuristics fine-tuned model on the DISFL-QA test-set. The overall heuristics trained model (ALL) is able to cover a significant performance drop from 61.64 to 82.27, an increase of 20.63 F1 points. However, this still is 7.32 F1 points short of the fluent performance.

Amongst the individual heuristics, we observe the following order of effectiveness w.r.t. performance on the HasAns cases: ENT > SQ > ADJ > VERB > ADV. One possible expla-

[5] We use the standard SQUAD evaluation script and mark a prediction as WrongAns iff F1(pred,gold)< 0.8.

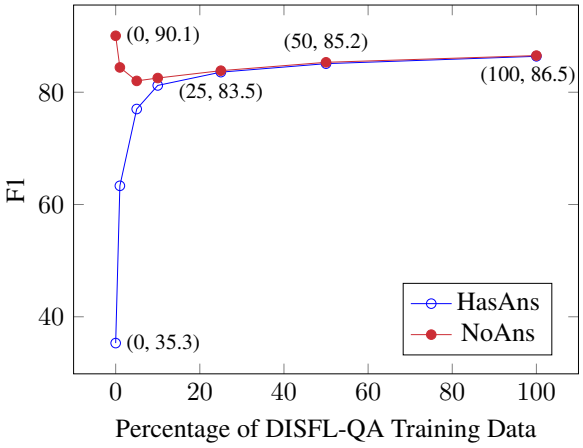

Figure 2: Few shot performance for different fraction of training data. We can see that performance on HasAns cases increases monotonically with increase in gold data. However, for the NoAns cases, the performance first takes a drop (compared to zero-shot) and then increases.

	HasAns F1	NoAns F1	Overall F1
Fluent (⋆)	91.38	87.67	89.59
Zero-Shot	35.21	90.06	61.64
Heuristics	78.86	85.96	82.27
Direct Supervision			
25% Data	83.58	83.84	83.71
+ Q → DQ	86.44	84.53	85.52
+ CQ → DQ	87.47	83.11	85.37
50% Data	85.09	85.33	85.20
100% Data	86.40	**86.53**	**86.46**
+ Q → DQ	86.95	85.73	86.33
+ CQ → DQ	**87.29**	85.22	86.29
Pipelined			
DQ → Q	87.65	**86.70**	**87.19**
CDQ → Q	**87.99**	86.02	87.04

Table 7: Performance on the test set of DISFL-QA when using gold human annotated data in training different components.

nation for SWITCH-ENT and SWITCH-Q being more effective is the fact that our original annotated dataset has a relatively high percentage of entity and interrogative correction.

4.2 Few Shot Performance

Next, we evaluate the performance of the models when we use a part of human annotated gold disfluent data for training: (i) direct end-to-end supervision, (ii) generation based data augmentation, and (iii) training disfluency correction models.

Direct Supervision (*k*-shot). In this setting, we pick a SQUAD-v2 T5 model and then perform a second round of fine-tuning with varying percentages of DISFL-QA gold training data. We experiment with $1, 5, 10, 25, 50$, and 100 percent of the total gold data.

Figure 2 shows the performance for the HasAns and NoAns cases as we increase the amount of training data. The HasAns performance increases gradually from 35.31 F1 points, in the zero-shot setting, to 86.40 F1 points with complete training data. Interestingly, for the NoAns cases, the performance first drops from 90.06 F1 points, in the zero-shot setting, to 82.02 F1 with 5% data and then monotonically increasing to 86.53 F1 with complete data. This can be attributed to the fact that the zero-shot models were under-predictive (high recall, low precision for <no asnwer>) due to lack of robustness to disfluent inputs.

Furthermore, Table 7 compares the performance of using the gold training data of DISFL-QA against the heuristics data. It shows that the models trained with disfluent data from DISFL-QA are able to cover a major gap in answerable slice, which wasn't possible with the heuristically generated data. Direct supervision bring an additional performance improvement of 4.19 F1 points over the heuristics.

Generation Based Data Augmentation. We use the T5 model for synthetically generating disfluent question from fluent question in the *text2text* framework. We use the training set of DISFL-QA to train the following generative models: (i) context-free generation (Q → DQ), and (ii) context-dependent generation (CQ → DQ) which use passage as well for generation.

Table 8 shows example generation from the two models. We observe that CQ → DQ is able to learn meaningful contextual disfluency generation, whereas Q → DQ can lead to non-meaningful or inconsistent disfluencies due to lack to context.

We then pick 5k random (question, answer) pairs from SQUAD training data and apply our generative model to produce disfluent training data for the QA models. Table 7 shows the performance of using data augmentation. We perform data augmentation under two different train data settings: (1) 25% data, and (2) 100% data. Interestingly, for the models trained on 25% train data + generated data, we observe a gain of 1.81 F1

Passage: ...*Whereas a* ***genome sequence*** *lists the order of every DNA base in a genome, a* ***genome map*** *identifies the landmarks. A genome map is less detailed than a genome sequence and aids in navigating around the genome* ...
Fluent Question : What does a genome map list the order of ? **T5 Q → DQ** : What is no what does a genome map list the order of ? **T5 CQ → DQ** : What does a genome sequence list the order of no sorry what does a genome map list the order of?
Passage: ...*The presence of* ***fat*** *in the small intestine produces hormones that stimulate the release of pancreatic lipase from the pancreas and* ***bile*** *from the liver which helps in* ...
Fluent Question : What is one molecule of fat ? **T5 Q → DQ** : What is one molecule of protein no fat ? **T5 CQ → DQ** : What is one molecule of bile no wait fat ?
Passage: ...*In 1964,* ***Nikita Khrushchev*** *was removed from his position of power and replaced with* ***Leonid Brezhnev****. Under his rule, the Russian SFSR* ...
Fluent Question : When did Leonid Brezhnev die ? **T5 Q → DQ** : When was the age of Leonid Brezhnev ? **T5 CQ → DQ** : When did Nikita Khrushchev *er* I mean Leonid Brezhnev die ?

Table 8: Example disfluent question (DQ) as generated by the Q → DQ and CQ → DQ T5 generative models for data augmentation. We observe that CQ → DQ generates meaningful disfluencies compared to context-free generation, the latter leading to irrelevant or inconsistent questions in some cases.

points (83.71 → 85.52) in the overall performance which is close to the absolute performance of using 50% gold data. However, for the setup with 100% gold data + generated data, we did not observe a similar improvement in the overall performance.

Pipelined: Disfluency Correction + QA. Unfortunately, existing disfluency correction models and datasets assume that fluent text is a subsequence of the disfluent one, and hence these approaches cannot solve disfluencies in DISFL-QA involving coreference. For fair comparison, we train a T5 generation model as a DISFL-QA specific disfluency correction model using the training set of DISFL-QA, with a simple DQ → Q and CDQ → Q T5 task formulation.

With this pipelined approach, we get further improvements with an overall F1 of 87.19 (Table 7), however, still lacking by ≈2.4 F1 points compared to the fluent dataset. This shows that such complex cases require better modeling, preferably in an end-to-end setup.

5 Related Work

5.1 Disfluency Correction

The most popular approach in literature poses disfluency correction as a sequence tagging task, in which the fluent version of the utterance is obtained by identifying and removing the disfluent segments (Zayats et al., 2014; Ferguson et al., 2015; Zayats et al., 2016; Lou and Johnson, 2017; Jamshid Lou and Johnson, 2020; Wang et al., 2020). . Traditional disfluency correction models use syntactic features (Honnibal and Johnson, 2014), language models (Johnson et al., 2004; Zwarts and Johnson, 2011), discourse markers (Crible, 2017), or prosody-based features for learning (Zayats and Ostendorf, 2019; Wang et al., 2017) while recent disfluency correction models largely utilize pre-trained neural representations (Lou et al., 2018). Most of these models depend on human-annotated data. As a result, recently, data augmentation techniques have been proposed (Yang et al., 2020; McDougall and Duckworth, 2017) to alleviate the strong dependence on labeled data. However, the resulting augmented data either via heuristics (Wang et al., 2020) or generation models (Yang et al., 2020) is often limited in terms of disfluencies types and may not well capture natural disfluencies in daily conversations.

5.2 Question Answering Under Noise

In the QA literature, our work is related to two threads that aim to improve robustness of QA models: (i) QA under adversarial noise, and (ii) noise arising from speech phenomena.

Prior work on adversarial QA have predominantly generated adversaries automatically (Zhao et al., 2018), which are verified by humans to ensure semantic equivalence (i.e. answer remains same after perturbation). For instance, Ribeiro et al. (2018) generated adversaries using para-

phrasing, while Mudrakarta et al. (2018) perturbed questions based on attribution. Closest work to ours is Jia and Liang (2017), who modified SQuAD to contain automatically generated adversarial sentence insertions.

Our work is more closely related to prior work on making NLP models robust to noise arising from speech phenomena. Earlier work (Surdeanu et al., 2006; Leuski et al., 2006) have built QA models which are robust to disfluency-like phenomenon, but they were limited in the corpus complexity, domain, and scale. Recently there has been renewed interest in constructing audio enriched versions of existing NLP datasets, for example, the Spoken-SQuAD (Li et al., 2018) and Spoken-CoQA (You et al., 2020) with the aim to show the effect of speech recognition errors on QA task. However, since collecting audio is challenging, another line of work involves testing the robustness of NLP models to ASR errors in transcribed texts containing synthetic noise using TTS → ASR technique (Peskov et al., 2019; Peng et al., 2020; Liu et al., 2020; Ravichander et al., 2021). Our work suggests a complementary approach to data collection to surface a specific speech phenomenon that affects NLP.

6 Conclusion

This work presented DISFL-QA, a new challenge set containing contextual semantic disfluencies in a QA setting. DISFL-QA contains diverse set of disfluencies rooted in context, particularly a large fraction of corrections and restarts, unlike prior datasets. DISFL-QA allows one to directly quantify the effect of presence of disfluencies in a downstream task, namely QA. We analyze the performance of models under varying when subjected to disfluencies under varying degree of gold supervision: zero-shot, heuristics, and k-shot.

Large-scale LMs are not robust to disfluencies. Our experiments showed that the state-of-the-art pre-trained models (BERT and T5) are not robust when directly tested on disfluent input from DISFL-QA. Although a naturally occurring phenomenon, the noise introduced by the disfluent transformation led to a non-answerable behavior at large.

Contextual heuristics partially recover performance. We derived heuristics, in attempt to resemble the contextual nature of DISFL-QA, by introducing semantic distractors based on NER, POS, and other questions. In our experiments, we found that heuristics are effective in: (1) confusing the models in zero-shot setup, and (2) partially recovering the performance drop on DISFL-QA with fine-tuning. This indicates that the heuristics might be capturing some key aspects of DISFL-QA.

Efficacy of gold training data. We use the gold data for supervising various models: (i) end-to-end QA model, (ii) disfluency correction, and (iii) disfluency generation (for data augmentation). For all the experiments, gold supervision outperforms heurisitics' supervision significantly. Furthermore, we observed that in a low resource setup generation based data augmentation can match the performance of a high resource modeling setup.

7 Discussion

While DISFL-QA aims to fill a major gap between speech and NLP research community, understanding *disfluencies* holistically requires the following:

General disfluencies focused NLP research. We believe understanding of disfluencies is a key ingredient for enabling natural human-machine communication in the near future, and call upon the NLP community to devise generalized few-shot or zero-shot approaches to effectively handle disfluencies present in input to NLP models, without requiring task specific disfluency datasets.

Constructing datasets for spoken problems. We would also like to bring attention to the fact that being a speech phenomenon, a spoken setup would have been an ideal choice for disfluencies dataset. This would have accounted for higher degree of confusion, hesitations, corrections, etc. while recalling parts of context on the fly, which otherwise one may find hard to create synthetically when given enough time to think.

However, such a spoken setup is extremely tedious for data collection mainly due to: (i) privacy concerns with acquiring speech data from real world speech transcriptions, (ii) creating scenarios for simulated environment is a challenging task, and (iii) relatively low yield for cases containing disfluencies. In such cases, we believe that a targeted and purely textual mode of data collection can be more effective both in terms of cost and specificity.

References

Eugene Charniak and Mark Johnson. 2001. Edit Detection and Parsing for Transcribed Speech. In *Proc. of NAACL*.

Ludivine Crible. 2017. Discourse Markers and (Dis)fluency in English and French: Variation and Combination in the DisFrEn Corpus. *International Journal of Corpus Linguistics*.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In *Proc. of NAACL*.

James Ferguson, Greg Durrett, and Dan Klein. 2015. Disfluency Detection with a Semi-Markov Model and Prosodic Features. In *Proc. of NAACL*.

John J Godfrey, Edward C Holliman, and Jane McDaniel. 1992. SWITCHBOARD: Telephone Speech Corpus for Research and Development. In *Proc. of ICASSP*.

Matthew Honnibal and Mark Johnson. 2014. Joint Incremental Disfluency Detection and Dependency Parsing. *Transactions of the Association for Computational Linguistics*.

Paria Jamshid Lou, Peter Anderson, and Mark Johnson. 2018. Disfluency Detection using Auto-Correlational Neural Networks. In *Proc. of EMNLP*.

Paria Jamshid Lou and Mark Johnson. 2017. Disfluency Detection using a Noisy Channel Model and a Deep Neural Language Model. In *Proc. of ACL*.

Paria Jamshid Lou and Mark Johnson. 2020. Improving Disfluency Detection by Self-Training a Self-Attentive Model. In *Proc. of ACL*.

Robin Jia and Percy Liang. 2017. Adversarial Examples for Evaluating Reading Comprehension Systems. In *Proc. of EMNLP*.

Mark Johnson and Eugene Charniak. 2004. A TAG-based noisy-channel model of speech repairs. In *Proc. of ACL*.

Mark Johnson, Eugene Charniak, and Matthew Lease. 2004. An Improved Model for Recognizing Disfluencies in Conversational Speech. In *Proc. of Rich Transcription Workshop*.

Anton Leuski, Ronakkumar Patel, David Traum, and Brandon Kennedy. 2006. Building Effective Question Answering Characters. In *In Proc. of SIGdial Workshop on Discourse and Dialogue*.

Chia-Hsuan Li, Szu-Lin Wu, Chi-Liang Liu, and Hung yi Lee. 2018. Spoken SQuAD: A Study of Mitigating the Impact of Speech Recognition Errors on Listening Comprehension. In *Proc. of Interspeech*.

Jiexi Liu, Ryuichi Takanobu, Jiaxin Wen, Dazhen Wan, Hongguang Li, Weiran Nie, Cheng Li, Wei Peng, and Minlie Huang. 2020. Robustness Testing of Language Understanding in Task-Oriented Dialog. *arXiv preprint arXiv:2012.15262*.

Paria Jamshid Lou, Peter Anderson, and Mark Johnson. 2018. Disfluency Detection using Auto-Correlational Neural Networks. In *Proc. of EMNLP*.

Paria Jamshid Lou and Mark Johnson. 2017. Disfluency Detection using a Noisy Channel Model and a Deep Neural Language Model. In *Proc. of ACL*.

Kirsty McDougall and Martin Duckworth. 2017. Profiling fluency: An Analysis of Individual Variation in Disfluencies in Adult Males. *Speech Communication*.

Pramod Kaushik Mudrakarta, Ankur Taly, Mukund Sundararajan, and Kedar Dhamdhere. 2018. Did the Model Understand the Question? In *Proc. of ACL*.

Baolin Peng, Chunyuan Li, Zhu Zhang, Chenguang Zhu, Jinchao Li, and Jianfeng Gao. 2020. RADDLE: An Evaluation Benchmark and Analysis Platform for Robust Task-oriented Dialog Systems. *arXiv preprint arXiv:2012.14666*.

Denis Peskov, Joe Barrow, Pedro Rodriguez, Graham Neubig, and Jordan Boyd-Graber. 2019. Mitigating Noisy Inputs for Question Answering. In *Proc. of Interspeech*.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2020. Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer. *In Proc. of JMLR*.

Pranav Rajpurkar, Robin Jia, and Percy Liang. 2018. Know What You Don't Know: Unanswerable Questions for SQuAD. In *Proc. of ACL*.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. SQuAD: 100,000+ Questions for Machine Comprehension of Text. In *Proc. of EMNLP*.

Abhilasha Ravichander, Siddharth Dalmia, Maria Ryskina, Florian Metze, Eduard Hovy, and Alan W Black. 2021. NoiseQA: Challenge Set Evaluation for User-Centric Question Answering. In *In Proc. of EACL*.

Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. 2018. Semantically Equivalent Adversarial Rules for Debugging NLP models. In *Proc. of ACL*.

Elizabeth Shriberg. 1996. Disfluencies in Switchboard. In *Proc. of ICSLP*.

Elizabeth Ellen Shriberg. 1994. *Preliminaries to a Theory of Speech Disfluencies*. Ph.D. thesis.

Mihai Surdeanu, David Dominguez-Sal, and Pere R Comas. 2006. Design and Performance Analysis of a Factoid Question Answering System for Spontaneous Speech Transcriptions. In *Proc. of ICSLP*.

Shaolei Wang, Wangxiang Che, Qi Liu, Pengda Qin, Ting Liu, and William Yang Wang. 2020. Multi-Task Self-Supervised Learning for Disfluency Detection. In *Proc. of AAAI*.

Shaolei Wang, Wanxiang Che, Yue Zhang, Meishan Zhang, and Ting Liu. 2017. Transition-Based Disfluency Detection using LSTMs. In *Proc. of EMNLP*.

Jingfeng Yang, Diyi Yang, and Zhaoran Ma. 2020. Planning and Generating Natural and Diverse Disfluent Texts as Augmentation for Disfluency Detection. In *Proc. of EMNLP*.

Chenyu You, Nuo Chen, Fenglin Liu, Dongchao Yang, and Yuexian Zou. 2020. Towards Data Distillation for End-to-end Spoken Conversational Question Answering. *arXiv preprint arXiv:2010.08923*.

Vicky Zayats and Mari Ostendorf. 2019. Giving Attention to the Unexpected: Using Prosody Innovations in Disfluency Detection". In *Proc. of NAACL*.

Vicky Zayats, Mari Ostendorf, and Hannaneh Hajishirzi. 2014. Multi-domain disfluency and repair detection. In *Proc. of Interspeech*.

Vicky Zayats, Mari Ostendorf, and Hannaneh Hajishirzi. 2016. Disfluency Detection Using a Bidirectional LSTM. In *Proc. of Interspeech*.

Victoria Zayats, Trang Tran, Richard A. Wright, Courtney Mansfield, and Mari Ostendorf. 2019. Disfluencies and Human Speech Transcription Errors. In *Proc. of Interspeech*.

Zhengli Zhao, Dheeru Dua, and Sameer Singh. 2018. Generating Natural Adversarial Examples. In *Proc. of ICLR*.

Simon Zwarts and Mark Johnson. 2011. The Impact of Language Models and Loss Functions on Repair Disfluency Detection. In *Proc. of ACL*.