DIFFQG: Generating Questions to Summarize Factual Changes

Jeremy R. Cole[1*]   **Palak Jain**[1*]   **Julian Martin Eisenschlos**[1]
Michael J.Q. Zhang[3]   **Eunsol Choi**[3]   **Bhuwan Dhingra**[1,2]
[1] Google Research   [2] Duke University   [3] The University of Texas at Austin
`{jrcole,palakj,eisenjulian,bdhingra}@google.com`
`{mjqzhang,eunsol}@utexas.edu`

Abstract

Identifying the difference between two versions of the same article is useful to update knowledge bases and to understand how articles evolve. Paired texts occur naturally in diverse situations: reporters write similar news stories and maintainers of authoritative websites must keep their information up to date. We propose representing factual changes between paired documents as question-answer pairs, where the answer to the same question differs between two versions. We find that question-answer pairs can flexibly and concisely capture the updated contents. Provided with paired documents, annotators identify questions that are answered by one passage but answered differently or cannot be answered by the other. We release DIFFQG which consists of 759 QA pairs and 1153 examples of paired passages with no factual change. These questions are intended to be both unambiguous and information-seeking and involve complex edits, pushing beyond the capabilities of current question generation and factual change detection systems. Our dataset summarizes the changes between two versions of the document as questions and answers, studying automatic update summarization in a novel way.

1 Introduction

Given a pair of statements, how can we identify the difference in their information content? This problem has existed in different forms across NLP research, such as *recognizing textual entailment* (Dagan et al., 2010) and *natural language inference* (Bowman et al., 2015). The initial focus of this type of research was finding the logical implication relations between sentences.

More recently, specialized entailment-like resources and models have been applied to fact verification (Thorne et al., 2018b) with applications to science, education and journalism. This trend has

---

[*] Equal Contribution.

Figure 1: DIFFQG consists of paired Wikipedia passages that correspond to factual edits. The goal is to generate a discriminating question given an answer span such that the question is answerable by one of the passages but not the other or yields different answers.

exposed the limited transfer between logical entailment and general factual change detection (Thorne et al., 2018a) as well as the need for interpretable models for this task (Kumar and Talukdar, 2020).

Wikipedia revisions across time provide a large scale and highly available source of sentence pairs, leading to new resources such as WIKIATOMICEDITS (Faruqui et al., 2018) and VITAMINC (Schuster et al., 2021). However, prior work is limited to minimal changes that concern only a single factual addition or change. We introduce DIFFQG, a manually annotated dataset spanning changes over multiple years. DIFFQG consists of paired passages with complex factual changes including multiple additions and deletions within the same example. Additionally, it provides a way to interpret the prediction in the form of a discriminative question-answer pair that identifies the change.

Question-answer pairs provide a semi-structured summary of a change: more flexible than knowledge graph triples and more useful than free-form text. For instance, question-answer pairs can represent different types of updates: a new prime minister may update an answer, while a new type of

3088

minister would add an entirely new question.

Question generation (QG) is a new NLP task that consists of generating a question that a provided document answers. There are various successful applications of this approach, including augmenting datasets to train question answering systems (Duan et al., 2017; Lewis et al., 2021), capturing implicit information written about text (Pyatkin et al., 2021), and building soft knowledge bases (Chen et al., 2022). Previous work in QG treated the underlying passages as static (Lewis et al., 2021), while real life documents are constantly updated (Dhingra et al., 2022). As the source corpus is updated, new question-answer pairs must be added and existing ones must be updated.

DIFFQG thus addresses two challenges simultaneously: providing an interpretable summarization of factual changes and updating soft knowledge bases consisting of question answer pairs. We hope that this dataset can also help evaluate the quality of QG models in producing natural, semantically correct, unambiguous, and information-seeking questions. The dataset and code for our experiments will be open sourced.[1]

Our contributions are the following:
(a) We introduce DIFFQG, an expert-annotated *evaluation* dataset that consists of questions that summarize the difference between two passages. To the best of our knowledge, no prior dataset exists that covers such long and complicated edits.
(b) We propose a set of metrics that can be used to measure improvements in question generation or factual change detection.
(c) We evaluate a comprehensive set of baselines that surface the shortcomings of current systems.

2 DIFFQG Task

The goal of DIFFQG is to capture how two similar passages differ from each other using question-answer pairs. In particular, given a base passage x_b and a target passage x_t, where x_t and x_b are different versions of the same article, we aim to generate discriminating questions Q_t. For each $q_t \in Q_t$, the information to deduce the corresponding answer span $a_t \in A_t$ must be missing in x_b. To limit the scope of possible questions, each answer span $a_t \in A_t$ must be a substring of x_t. While a_t could also be a substring of x_b, x_b must be missing the required information to deduce a_t is the correct

answer. Alternatively, there could be a corresponding answer span a_b, which is the answer resulting from answering q_t with x_b. Note that we consider paraphases of a_t, such as lexicalizing numbers and using alternate entity names, as equivalent answers.

This discriminating question has certain additional requirements: it should be seeking factual information and stand-alone (Choi et al., 2021) (i.e., interpretable when presented by itself without the passage). It is possible that no such discriminating question can be written. The annotators only mark that there is no factual change when they are fairly confident that there is no new information about the answer span in the target passage.

Consider the following example:

- x_b = John Doe won two gold medals at the Olympics in 2012.

- x_t = John Doe won a gold medal at the Olympics in 2012.

Annotators are informed that the goal of the process is to collect *disambiguated* and *information-seeking* queries that can be answered with one passage but not with the other. By *disambiguated* queries, we mean queries that refer to roughly a single answer without any context. For instance, "Who won two gold medals in the 2012 Olympics?" could refer to several different people, and questions of the form "How many medals did he win in the 2012 Olympics?" are not answerable at all without the presence of the John Doe passage.

Information-seeking queries are ones where the questioner would not need to know the answer in advance for the question to make sense. This is related to the original goals of Natural Questions (Kwiatkowski et al., 2019) and corresponds to the *Cranfield*-style questions described by Rodriguez and Boyd-Graber (2021). As an example, "What did Al Capone's mother do for a living?" seems like an information-seeking query. On the other hand, "Which Italian-American gangster's mother was a seamstress?" does not: why would the questioner assume that such a person even exists unless they already knew the answer? We describe the annotation process to acquire such discriminating question set in the next section.

3 Data Collection

Collecting such discriminating questions is a nontrivial process. Thus, we introduce a staged annotation process with expert annotators (the authors

---

[1] https://github.com/google-research/language/tree/master/language/diffqg

of this paper) and use a question generation model to aid annotation. We describe our process below (visualized in Figure 2).

3.1 Input Passage Pair Selection

First, we extract the Wikipedia pages for entities from the Natural Questions (NQ) training set (Kwiatkowski et al., 2019). In particular, we find the pages for Wikipedia snapshots between the years 2008 and 2020. After sampling a base document, we find the version of that document one year later and use this as the target document. Using the two documents as a corpus, we compute cosine similarity between the TF-IDF vectors over each sentence pair and pick the pair with the highest similarity. Sentence pairs with similarity either greater than 0.8 or less than 0.25 are discarded. In order to retain meaningful changes, we ensure at least one noun or number is edited and up-sample instances where either a named entity or at least five tokens have been edited.

This process thus focuses on edits accumulated over a year and consists of changes ranging from five to twenty tokens, making these semantically richer and more widely applicable than existing factual change detection datasets.

3.2 Seed QA Pair Generation

Each target passage has a very large number of possible answer spans; for convenience, we restrict them to only noun phrases identified using the Berkeley Neural Parser (Kitaev and Klein, 2018). To increase annotation speed, each example starts with a *seed question* that is generated by a question generation model from the target passage and answer span. In particular, we use a T5-XXL model (Raffel et al., 2020) that has been finetuned on the SQuAD dataset (Rajpurkar et al., 2016).

3.3 Annotation Process

DIFFQG annotation was done in three phases by six expert volunteers. First, annotators are given the paired passages described above along with the answer span and seed question, which corresponds to one *example candidate*. Then, they label each example candidate with one of the five options:

Accept The seed question follows all requirements for discriminating questions as is.

Context The seed question asks about the appropriate topic but is not answerable outside of the context of the passage. For instance, questions like

"What did he win?" or "Where were the Olympics held?" both lack context in order to answer the question successfully.

Edit The example candidate answer has a discriminating question, but the question is different than the seed question. Sometimes, this is because the seed question does not capture the new information contained in the passage; other times, the seed question is simply nonsense.

Reject This example candidate has no valid discriminating question. In other words, these are negative examples. Sometimes, the target passage contains no new information at all; however, it may contain new information about other answer spans but not the one in the example candidate. In our previous John Doe example, there is no new information about "the Olympics", except indirectly.

Skip It is unclear if there is a valid discriminating question for this example candidate. This could be due to awkward or cumbersome answer spans: for instance "two gold medals at the Olympics in 2012." Alternatively, it could seem unclear if there is new information about an answer span due to its indirect relationships with other entities. Finally, it could be difficult to write an information-seeking question even though there is obviously new information: for instance, writing a question with the answer span "John Doe" in the previous example.

Each example candidate is considered by two annotators. Unless both annotators agree to Add, Reject, or Skip, a third annotator decides. In examples where one annotator chose Context or Edit, the third annotator is responsible for writing the correct question according to the guidelines. If one annotator chose Add or Reject and the other skipped, the third annotator can confirm the Add or Reject or also skip if they cannot decide. See Appendix A.2 for the annotation interface.

3.4 Question Writing Guidelines

Note that writing a single, context-free, and information-seeking question that summarizes the difference between the two passages can be challenging. In cases where it seemed impossible, annotators are encouraged to skip the example. For cases where additional Context was needed, annotators are encouraged to add as much context without sacrificing fluency, so that the question can be answered without awareness of the source passage. When an annotator writes a question from scratch
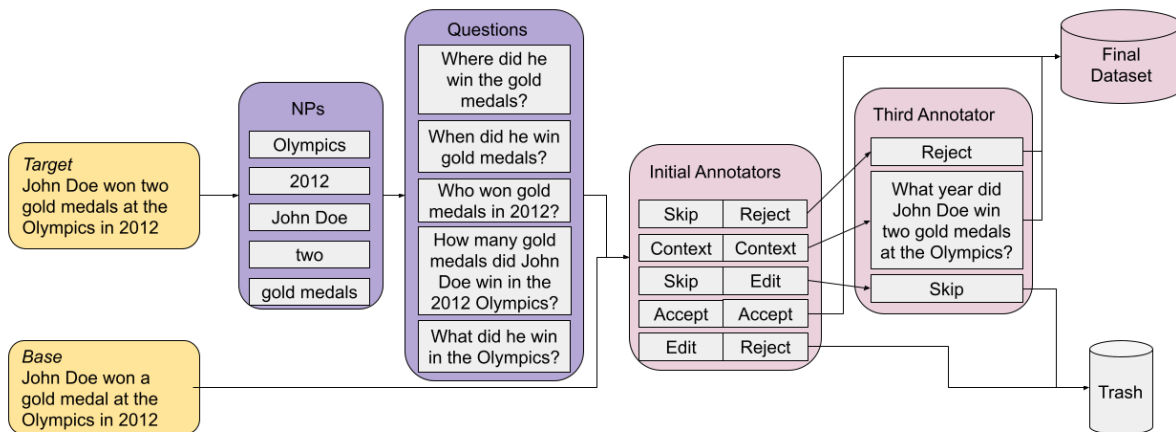
Figure 2: DIFFQG annotation process. Noun phrases are extracted from the target passage and a question generation model seeds initial questions. Annotators decide if the generated questions serve as satisfactory discriminating questions (Accept), must be edited (Context, Edit), or contain no new factual information (Reject). After the first phase, a third annotator resolves indecision (Skip), leaving us with a set of questions and negative examples. If the original two annotators disagree or the third annotator cannot resolve indecision, the example is discarded.

in the Edit case, they are encouraged to think of a question that either would have a different answer, be unanswerable, or have a false precondition if posed against the base passage. While conditioning on a single answer span reduces ambiguity, the task is still ambiguous, which is unavoidable when handling large and complex edits.

	Passages	Answers	Avg. Edited tokens
w/ change	391	759	12.9
w/o change	478	1153	14.1
Total	672	1912	13.7

Table 1: Dataset statistics for DIFFQG. Edited tokens represents the average tokens added or removed in a given passage pair.

3.5 Data Statistics

Our initial annotation process starts with $8,530$ example candidates drawn from 999 passage pairs. Annotators skipped nearly 75% of the example candidates, leaving $1,912$ examples. Of those, roughly 40%, or 759, had a factual change and thus a discriminating question written about them, leaving 1153 negative examples. Of the spans where a factual change was detected, annotators modified the question in 65%, or 494, of the examples: 45% are labeled as Context and 20% as Edit. Detailed dataset statistics can be found in Table 1.

Note that on all cases where a question was accepted as is or considered a negative example, at least two annotators agreed on that rating. However, human written questions are not verified; both

annotators agree that there exists a discriminating question but not necessarily what it is. To address this, we evaluate a small set of fifty questions and found that a second annotator would write an equivalent question around 85% of the time.

4 Motivation

In the previous section, we described DIFFQG and its annotation procedure. As mentioned, the purpose of DIFFQG is to detect and describe factual changes. In particular, DIFFQG is a rough measurement of a model's ability to automatically construct a database of question-answer pairs that encapsulate the changes. There are many possible formats that could be used as an alternative to summarize factual changes, such as paragraphs, knowledge base triples, or individual claims.

While paragraphs can contain nuance, they lack atomicity. It is thus difficult to tell what exactly changed or otherwise compare two changes to each other. This makes them less useful as a database.

On the other hand, knowledge base triples are limiting in the types of factual changes that can be described: regardless of the exact setup, the nodes and relations come from some form of fixed vocabulary that may require discarding interesting changes. For instance, changes related to a set of entities, date ranges, various numbers, or abstract information may all be challenging.

Another alternative method would be a list of claims, similar to Vitamin-C (Schuster et al., 2021). This method is also atomic and more flexible than

knowledge base triples. However, question-answer pairs have a few advantages. First, question-answer pairs are semi-structured information, forming a loose key-value pair. Factual edits may change the answer to an existing question or add information corresponding to an entirely new information, requiring a new question. Conversely, claims are more difficult to relate to each other.

Finally, question-answer pairs are interesting because question answering is interesting. Previous work has seen the use of a database of question-answer pairs as a method to improve question answering performance (Lewis et al., 2021). A good method for automatically creating and updating such a database thus seems quite useful. As factual corpora change over time, we envision constructing such a database to require iterative updates.

5 Metrics

DIFFQG can be used to measure performance on three related tasks.

Factual Change Detection Given an example consisting of a base passage, target passage, and answer span, the goal is to determine whether there exists a valid differentiating question. In other words, whether there is new information about this answer span that is present in the target passage when compared to the base passage. To measure this, we report accuracy, precision, recall and F1 score over the existence of a differentiating question in our annotations. Note that always predicting no change achieves 60.3% accuracy but 0% F1, but random guessing corresponds to 44.1% F1.

Discriminating Question Generation Given a target passage and answer span, write a specific, unambiguous and information-seeking query that can be answered with the target passage. To measure this, we compare machine generated questions to those that humans verified, edited, or hand wrote. We use two model-free metrics Rouge-1 and Rouge-L (Lin, 2004) which measure the token-level overlap and longest subsequence overlap of the questions, respectively. We also consider two model-based metrics, BLEURT (Sellam et al., 2020), which is a learned evaluation for text similarity based on BERT (Devlin et al., 2019), and a query similarity model (Reimers and Gurevych, 2019) trained on Quora Question Pairs [2].

Note that we evaluate discriminating question generation despite using a question generation model in our annotation procedure. Note that all of these questions are reviewed by humans and only the very fluent ones are kept. As question generation models vary in which of their productions are very fluent, this set is less trivial than it would initially appear. Nonetheless, we also separate human-written or edited questions and evaluate that set independently.

Full System This is the overall measure of performance on DIFFQG. We reuse the metrics from discriminating question generation, using 0.0 for BLEURT, ROUGE-1, ROUGE-L, and Query Similarity if the factual change detection is incorrect.

6 Methods

As mentioned, DIFFQG can be thought of as a composition of two tasks: factual change detection and discriminating question generation. Our simple baseline systems thus treat this as a pipeline, first predicting whether or not there is a factual change and then generating a discriminating question if there is. We also present baseline models that solve both tasks jointly with a single prediction. Our methods are illustrated in Figure 3.

Note that none of our methods use any part of DIFFQG as training data, as the dataset is only intended to be used for evaluation. Models are instead trained on larger existing datasets for question generation and factual change detection.

6.1 Factual Change Detection

We propose five baselines based on answer equivalence or both question and answer equivalence.

Answer Equivalence Baselines

Our trivial baseline (*Overlapping Answer*) classifies an example as having a factual change if and only if the answer span is not present in the base passage. The span is normalized before looking for token overlap with the passage.

Our simple model-based baseline is similar but uses an Answer Equivalence model (Bulian et al., 2022). It compares the target answer span against all valid base answer spans, finding a factual change if it does not match any of them. The Answer Equivalence model additionally takes as input a candidate question for each answer span.
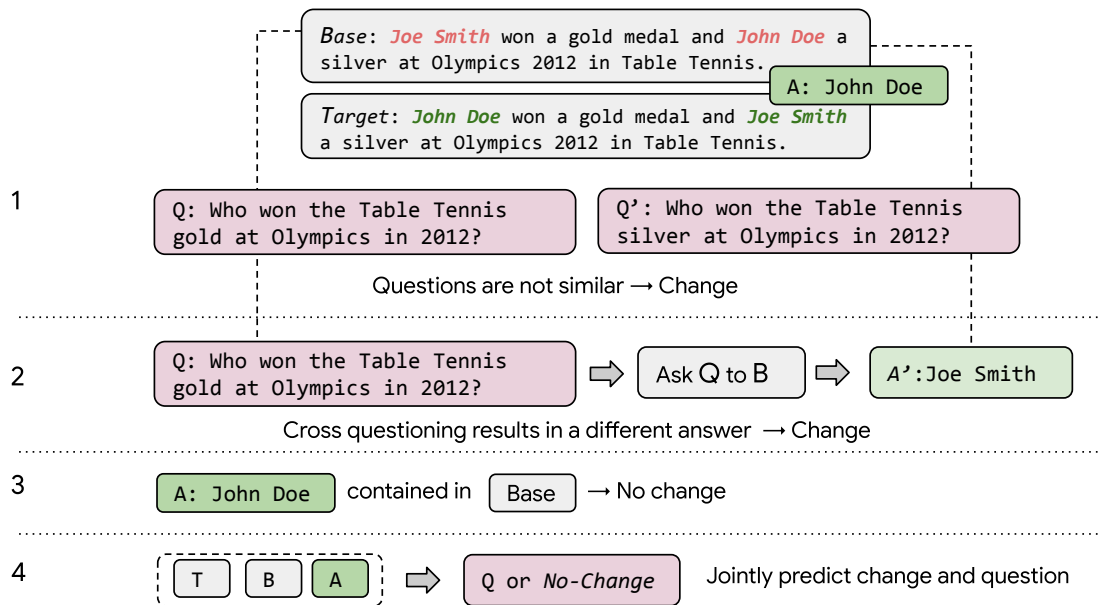
Figure 3: Methods for factual change detection on DIFFQG: (1) Question similarity (2) Cross-Questioning. The QA-equivalence method combines (1) and (2), deciding it is a Change only when both the systems find a Change. (3) Overlapping Answers (4) Language model that jointly learns the task of factual change detection and discriminating question generation, decoding the discriminating question or a special token indicating no change.

Question-Answer Equivalence Baselines

The previous methods only consider the answer span, ignoring the context. Here, we consider methods that also use a question generation model on the passages and answer span to determine if there is new information.

For the first method, we find base answer spans equivalent to the target answer span using the *Overlapping Answer* method. Then, we want to see if the questions generated from those answers would also be equivalent in both passages. To do so, we use a T5-XXL ([Raffel et al., 2020](#)) model trained on Quora Question Pairs [3] to predict whether the pair of questions is "duplicate" or "not duplicate". If the question is not a duplicate, then we consider this example to have a factual change. Thus, answer spans present in both passages but with different contexts could now be identified as having a factual change. This will increase the recall of the *Overlapping Answer* method.

The second approach adds a cross questioning filter (*Cross-Q*). Given a candidate question generated from target passage, we attempt to answer the question with the base passage using a reading comprehension model. We train a T5-XXL model on SQuAD v2 ([Rajpurkar et al., 2018](#)) question-answering dataset to take the passage and ques-

tion as input and output the answer. If the model predicts no answer or a different answer from the target span, we classify the example as having a factual change. Finally, the *QA-equivalence* method combines both the query similarity model and cross question model to boost precision. In this case, we consider an example to have a factual change only when both methods determine a factual change.

6.2 Question Generation

Each of our factual change detection baselines is then combined with a question generation model. We use a similar T5-XXL model finetuned on SQuAD as described in [Section 3.2](#). Unsurprisingly, the model we use to seed the questions can do well on the questions that it wrote originally; however, this is an unfair baseline. Thus, we additionally test a version of the model that is sampled and also a retrained version using a different seed. We also test training a similar model trained on Natural Questions ([Kwiatkowski et al., 2019](#)).

6.3 Joint systems

Many of the techniques described [Section 6.1](#) are inefficient, requiring multiple runs of various models. For instance, the Query Similarity method requires one model run for each answer span in the base passage per example, which corresponds to quadratic runs for each pair of passages. We

---

[3]https://www.kaggle.com/c/quora-question-pairs

also explore methods that can directly compare the base and target passage without the need for any intermediate steps. These methods instead jointly detect if there is a factual change and generate a discriminating question.

Finetuning on Silver Data

We mine additional pairs of Wikipedia passages using the same process as in Section 3.1. We then identify every possible answer span from the target passage. We create silver training examples for the factual change detection component of the task by labeling each target answer span using our best heuristic method, *QA-equivalence*.

We then convert these labels into a text-to-text task. For each example with a factual change, we use the question generated by the SQuAD model (Section 6.2) as the target. For questions without a factual change, we use "None" as the target. The input to the model is the concatenation of the base and target passages with the target answer span marked by a special token.

The model is a T5-XXL initialized from the same question generation model as the model that produced the original questions.

Finetuning on VITAMINC

The VITAMINC task (Schuster et al., 2021) also has a factual change detection component. We sample negatives from the VITAMINC Revision Flagging dataset, using negative examples with a random noun phrase chosen as the answer span.

For positive examples, we need to identify a specific answer span that contains a factual change as well as the corresponding discriminating question. While there is no direct counterpart of this task in VITAMINC, the Fact Verification task is somewhat similar. The dataset consists of an evidence e, a simplified claim c supporting e, a companion edited sentence e' and an edited-claim c' refuting e and supporting e'. An answer span a is identified based on the token-level diff between (c, c') and a question generated from c conditioned on a using the question generation model in Section 6.2. Because c is a simple sentence, we anecdotally find the generated questions to be of high quality.

The dataset (e, e', a) is converted to a text-to-text task and used to finetune a T5-XXL model following the same steps as above. Note that a model trained on an equal amount of positives and negatives yielded poor performance on DIFFQG. In our final VITAMINC silver dataset, we used only 10%

Model	Acc	P	R	F1
Random	50.0	39.5	50.0	44.1
Overlapping Ans	82.1	79.1	74.6	76.7
Answer Equivalence	81.1	84.6	64.2	73.0
Query Similarity	77.7	65.4	93.3	76.9
Cross-Q	76.9	65.9	86.8	74.9
QA-equivalence	**83.9**	76.7	85.5	**80.9**
FT on QA-equivalence	<u>82.5</u>	78.4	77.1	<u>77.7</u>
FT on VITAMINC	81.5	79.9	71.4	75.4

Table 2: Metrics for factual change detection. Note that none of these models have change detection training data and are instead verifying with other tasks or heuristics. The random baseline assumes guessing change or No change with equal probability. Acc=Accuracy,P=Precision,R=Recall,F1=F1 Score. **Bold** indicates the best model, second best model is <u>underlined</u>.

negatives to achieve a reasonable performance.

7 Results and Discussion

We present results separately for factual change detection, question generation and the full system. We also report results separately for the overall performance and the performance on only the subset of questions that are human written; those sentences labeled as Edit or Context in the annotation phase. Selected examples with model outputs are provided in Appendix A.1 to illustrate the capabilities and typical errors baseline.

7.1 Factual Change Detection

Table 2 compares the performance of various systems on the factual change detection task. We find that QA-equivalence performs better than heuristic baseline methods. In particular, it better handles cases where the answer span text is unchanged, but the surrounding context has changed. For example, in the passage *"On the New Hampshire Executive Council, Laconia is in the 1st District, represented by <ADD: Republican Joe Kenney> <DEL: Democrat Michael J. Cryans>."*, QA-equivalence correctly captures the new information associated with the answer span *"New Hampshire Executive Council"* in the form of the question *"What state council does Joe Kenney represent Laconia in?"* However, the method is prone to detecting spurious changes even when the passages have no semantic edit as illustrated in Appendix A.1.

The joint system finetuned on silver data from QA-equivalence does not seem to improve upon

QA-equivalence. While it seemingly benefits from the additional context, it still struggles with long and complex edits. However, this model only requires a single inference to do both tasks.

The VITAMINC trained model, despite having access to additional data, was also unable to improve on our baseline. VITAMINC style edits are substantially different than DIFFQG edits, generally only consisting of small changes. Thus, the model finetuned on VITAMINC performs poorly on large phrase changes or sentence refactors.

7.2 Question Generation

Model	R-1	R-L	QSim	BLRT
SQuAD-seed	71.7	74.9	66.3	74.1
SQuAD-sampled	59.6	63.2	57.8	65.6
SQuAD-retrained	58.0	61.7	58.9	65.6
NQ	20.4	39.6	22.9	41.5

Table 3: Variation in performance of question generation models on the positive subset of DIFFQG. We report three different versions of the SQuAD model, where the first model is the same as we used to seed the annotations. R-1=ROUGE-1, R-L=ROUGE-L, QSim=Query Similarity model-based accuracy, BLRT=BLEURT.

In Table 3, we compare our question generation baseline models on the subset of the positive examples. In Table 4, we examine the same models on the subset of those that are human written: examples with a change from Table 1.

Model	R-1	R-L	QSim	BLRT
SQuAD-seed	56.5	61.4	48.2	61.3
SQuAD-sampled	50.9	55.2	47.0	58.3
SQuAD-retrained	50.3	54.5	50.4	59.4
NQ	20.4	39.1	20.0	40.2

Table 4: Variation in performance of question generation models on human written questions of DIFFQG. The first model is the same as what we used to seed the questions. R-1=ROUGE-1, R-L=ROUGE-L, QSim=Query Similarity model-based accuracy, BLRT=BLEURT.

The primary goal of this evaluation is to test whether the questions directly produced by the seed model described in Section 3.2 are still useful for evaluating systems on DIFFQG. We find from sampling from that same model and from retraining with the same process (as described in Section 6.2)

that performance on the overall set degrades considerably. This suggests that unless someone had access to the same model, these questions that are human-verified but not human written can still be useful for evaluation. Nevertheless, the seed model can be thought of as a rough ceiling on current question generation performance on DIFFQG.

The human written questions (see Table 4) seem to be much more challenging for the question generation models to replicate. Performance degrades substantially: naturally it degrades the most for the seed model that wrote some of the questions in the overall dataset, which it should exactly match.

We note also that a question generation model finetuned on Natural Questions (Kwiatkowski et al., 2019) yields a significantly different question style than SQuAD. This is likely because SQuAD questions are originally generated from passages, while Natural Questions are more free form. In addition the Natural Questions model is found to hallucinate in numerous scenarios. This reflects on the poor performance of the Natural Questions-trained question generation model on DIFFQG.

As a caveat, the possible universe of questions written to summarize a factual change can be very large. While restricting to a single answer span reduces this space, we still find scenarios with multiple valid questions. Thus, there may be some disagreements where the model generates a completely valid question that is simply not the most pertinent one according to our annotators.

7.3 Full System

Full DIFFQG metrics are presented in Table 5 and include the two finetuned systems that are trained on VITAMINC and QA-equivalence, respectively, as well as two pipelined systems with factual change detection models attached to a question generation model. For the pipelined experiments, we use the retrained SQuAD model described in Section 6.2. We evaluate these models on the full DIFFQG as well as human written subset.

Overall, all of the systems are relatively close in performance. QA-equivalence works the best, with the finetuned version and simple heuristic model close behind, indicating substantial room for future innovation. On the human written subset, the performance drops significantly further highlighting the challenge of the human written questions.

Models		Factual Change Detection			Full System					
					All			Human written		
Change	QG	P	R	F1	R-L	Qsim	BLRT	R-L	Qsim	BLRT
Pipelined systems										
Overlapping Answer	SQuAD	79.1	74.6	76.7	71.3	70.8	<u>72.3</u>	40.6	38.1	43.6
QA-equivalence	SQuAD	76.7	85.5	**80.9**	71.4	70.6	**72.7**	46.8	43.7	**50.8**
Joint systems										
FT on QA-equivalence		78.4	77.1	<u>77.7</u>	71.2	70.0	<u>72.3</u>	42.7	36.6	<u>45.9</u>
FT on VITAMINC		79.9	71.4	75.4	69.9	68.1	71.1	36.3	32.2	39.9

Table 5: Full system performance of the pipelined and joint systems on DIFFQG. Note that the "All" component of the full system metric includes all of DIFFQG while the "Human written" portion includes only questions edited by the annotators. The pipelined systems use the retrained SQuAD model for their question generation component. **Bold** represents the best system, second best is <u>underlined</u>. R-L=Rouge-L, Qsim=Query similarity model based accuracy, BLRT=Bleurt, FT=finetuned

8 Related Work

Factual Edits Factual change detection has been of recent interest to the community. For instance, WIKIATOMICEDITS (Faruqui et al., 2018) rely on Wikipedia revisions to learn to discriminate factual edits. Closest to our work is VITAMINC (Schuster et al., 2021) which aims to generate a discriminating claim given a pair of edited sentences. However, both of these datasets primarily rely on smaller edits, frequently consisting of a single entity or number substitution. For instance, VITAMINC examples have a median of four token changes and WIKIATOMICEDITS examples have a median of two token changes. Moreover, these edits are easier to detect using heuristics such as noun or entity overlap. On the other hand, DIFFQG examples have a median of thirteen token changes that can involve multiple entity updates. Further, the surrounding contextual information for an entity could be updated even when the entity itself is present in both passages. This makes DIFFQG edits harder to summarize and substantially different than previous work; this is also observed in Section 7.1 where using VITAMINC training data to solve DIFFQG yields poor performance.

Recent work such as Fruit (Iv et al., 2022) and PEER (Schick et al., 2023) also operate on more complicated edits. Fruit generates updated sentences from a base passage given the new evidence in a Wikipedia article. PEER attempts to imitate the editing process using a sequence of planning steps. However, both of these primarily focus on generating the target update, while we focus on suc-

cinctly capturing the edited information. Further, the use of question generation as a device for discrimination is novel to the best of our knowledge.

Question Generation Question generation has been successfully applied to various purposes, including augmenting question answering systems (Duan et al., 2017; Lewis et al., 2021), capturing implicit information written about text (Pyatkin et al., 2021), and building soft knowledge bases (Chen et al., 2022). In this work, we apply question generation to the task of discriminating edited sentences. As far as we are aware, there is no prior work on evaluating question generation systems.

9 Conclusion

In this work, we introduce the DIFFQG task and dataset to evaluate the ability of NLP systems to summarize changes between two related passages via question generation. We present several heuristic and model baselines as well as a set of metrics to measure performance on the dataset. The DIFFQG task requires models to identify changes in factual relationships and ignore other stylistic edits. We find that existing approaches struggle under these conditions. Models trained to perform factual change detection and question generation jointly sometimes fail to understand even simple edits. We hope this work finds value in future research on this important problem.

Limitations

DIFFQG is relatively small, consisting of less than a thousand questions and less than two thousand

total examples. This makes us unable to provide a training set, limiting claims we can make about the difficulty of the task. Moreover, summarizing complex edits can have a large space of valid solutions. While using questions conditioned on an answer reduces this space, there's still room for ambiguity.

To make annotation easier, we use a question generation model; however, our goal is also to evaluate question generation models, complicating our story. Finally, most of the baselines we evaluate are some form of T5 (Raffel et al., 2020) model. It is possible that other model architectures could have solved this task more effectively.

Acknowledgements

References

Samuel R. Bowman, Gabor Angeli, Christopher Potts, and Christopher D. Manning. 2015. A large annotated corpus for learning natural language inference. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 632–642, Lisbon, Portugal. Association for Computational Linguistics.

Jannis Bulian, Christian Buck, Wojciech Gajewski, Benjamin Boerschinger, and Tal Schuster. 2022. Tomayto, tomahto. beyond token-level answer equivalence for question answering evaluation. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*.

Wenhu Chen, Pat Verga, Michiel de Jong, John Wieting, and William Cohen. 2022. Augmenting pretrained language models with qa-memory for open-domain question answering.

Eunsol Choi, Jennimaria Palomaki, Matthew Lamm, Tom Kwiatkowski, Dipanjan Das, and Michael Collins. 2021. Decontextualization: Making sentences stand-alone. *Transactions of the Association for Computational Linguistics*, 9:447–461.

Ido Dagan, Bill Dolan, Bernardo Magnini, and Dan Roth. 2010. Recognizing textual entailment: Rationale, evaluation and approaches. *Journal of Natural Language Engineering*, 4.

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.

Bhuwan Dhingra, Jeremy R. Cole, Julian Martin Eisenschlos, Daniel Gillick, Jacob Eisenstein, and William W. Cohen. 2022. Time-Aware Language Models as Temporal Knowledge Bases. *Transactions of the Association for Computational Linguistics*, 10:257–273.

Nan Duan, Duyu Tang, Peng Chen, and Ming Zhou. 2017. Question generation for question answering. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 866–874, Copenhagen, Denmark. Association for Computational Linguistics.

Manaal Faruqui, Ellie Pavlick, Ian Tenney, and Dipanjan Das. 2018. WikiAtomicEdits: A multilingual corpus of Wikipedia edits for modeling language and discourse. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 305–315, Brussels, Belgium. Association for Computational Linguistics.

Robert Iv, Alexandre Passos, Sameer Singh, and Ming-Wei Chang. 2022. FRUIT: Faithfully reflecting updated information in text. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3670–3686, Seattle, United States. Association for Computational Linguistics.

Nikita Kitaev and Dan Klein. 2018. Constituency parsing with a self-attentive encoder. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2676–2686, Melbourne, Australia. Association for Computational Linguistics.

Sawan Kumar and Partha Talukdar. 2020. NILE : Natural language inference with faithful natural language explanations. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8730–8742, Online. Association for Computational Linguistics.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, Kristina Toutanova, Llion Jones, Matthew Kelcey, Ming-Wei Chang, Andrew M. Dai, Jakob Uszkoreit, Quoc Le, and Slav Petrov. 2019. Natural questions: A benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:452–466.

Patrick Lewis, Yuxiang Wu, Linqing Liu, Pasquale Minervini, Heinrich Küttler, Aleksandra Piktus, Pontus Stenetorp, and Sebastian Riedel. 2021. PAQ: 65 million probably-asked questions and what you can do with them. *Transactions of the Association for Computational Linguistics*, 9:1098–1115.

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81.

Valentina Pyatkin, Paul Roit, Julian Michael, Yoav Goldberg, Reut Tsarfaty, and Ido Dagan. 2021. Asking it all: Generating contextualized questions for any semantic role. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 1429–1441, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, Peter J Liu, et al. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.*, 21(140):1–67.

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.

Nils Reimers and Iryna Gurevych. 2019. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics.

Pedro Rodriguez and Jordan Boyd-Graber. 2021. Evaluation paradigms in question answering. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 9630–9642, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Timo Schick, Jane A. Yu, Zhengbao Jiang, Fabio Petroni, Patrick Lewis, Gautier Izacard, Qingfei You, Christoforos Nalmpantis, Edouard Grave, and Sebastian Riedel. 2023. PEER: A collaborative language model. In *International Conference on Learning Representations*.

Tal Schuster, Adam Fisch, and Regina Barzilay. 2021. Get your vitamin C! robust fact verification with contrastive evidence. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 624–643, Online. Association for Computational Linguistics.

Thibault Sellam, Amy Pu, Hyung Won Chung, Sebastian Gehrmann, Qijun Tan, Markus Freitag, Dipanjan Das, and Ankur Parikh. 2020. Learning to evaluate translation beyond English: BLEURT submissions to the WMT metrics 2020 shared task. In *Proceedings of the Fifth Conference on Machine Translation*, pages 921–927, Online. Association for Computational Linguistics.

James Thorne, Andreas Vlachos, Christos Christodoulopoulos, and Arpit Mittal. 2018a. FEVER: a large-scale dataset for fact extraction and VERification. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 809–819, New Orleans, Louisiana. Association for Computational Linguistics.

James Thorne, Andreas Vlachos, Oana Cocarascu, Christos Christodoulopoulos, and Arpit Mittal, editors. 2018b. *Proceedings of the First Workshop on Fact Extraction and VERification (FEVER)*. Association for Computational Linguistics, Brussels, Belgium.

A   Appendix

A.1   Qualitative Examples

Examples from DiffQG dataset are illustrated in
Figure 4. The model outputs (success or failure)
from various systems are also provided alongside.

A.2   Annotation Interface

Refer Figure 5 and Figure 6 for annotation interface
of phase 1 and 2 respectively.

Figure 4: DIFFQG examples with predictions from various systems. The edited sentence is color coded with green for added tokens and red for deleted; the answer span is underlined. Additional context is omitted unless required for illustration (provided in gray). *No change* indicates there was no factual change for the example.

On the *New Hampshire Executive Council*, Laconia is in the 1st District, represented by **Republican Joe Kenney.** ~~Democrat Michael J. Cryans.~~
Ground Truth: What state council does Joe Kenney represent Laconia in
Overlapping Answer: No change **QA-equivalence**: What state council does Joe Kenney represent Laconia in **Ft on QA-equivalence**: No change

..triggering the Spanish–American War of 1898.. **In the Pacific, Cuba became independent while** the *United States* took possession of **Puerto Rico and Spain's Pacific colonies of** the ~~Spanish~~ Philippines and Guam.
Ground Truth: Which country took possession of Puerto Rico after the Spanish-American War of 1898
Overlapping Answer: No change **QA-equivalence**: Which country took possession of Puerto Rico after the Spanish-American War **Ft on QA-equivalence**: What country took possession of Puerto Rico after the Spanish-American

Dances with Wolves is a 1990 epic film which tells the story of a **United States cavalry officer from the** Civil War ~~era United States Lieutenant~~ who travels ~~to~~ **alone into** the ~~American~~ frontier **near** ~~to find~~ a ~~military post.~~ **_Sioux tribe_.**
Ground Truth: In Dances with Wolves, the protagonist travels close to which tribe
Overlapping Answer: Who did the cavalry officer come close to in the story **QA-equivalence**: Who did the cavalry officer come close to in the story **Ft on QA-equivalence**: No change

~~Additional music,~~ **So far Ryu and Chun Li each have one new costume each for $.99 and** '*Street Fighter II Arranged BGM*' can also be purchased to provide alternate in game audio.
Ground Truth: No change
Overlapping Answer: No change **QA-equivalence**: What can be purchased to change the in-game audio **Ft on QA-equivalence**: No change

DIFF 100: Susan Boyle (Id: 22378444_902516498_0)				
Base		**Target**		
Context: 2018. [SEP] Susan Magdalane Boyle (born 1 April 1961) is a Scottish singer who came to international attention when she appeared as a contestant on the TV programme Britain's Got Talent on 11 April 2009, singing "I Dreamed a Dream" from . [SEP] Her first album, I Dreamed a Dream, was released in November 2009 and became the UK's best-selling debut album of all time, beating the previous record held by Spirit by Leona Lewis.		**Context**: 2019. [SEP] Susan Magdalane Boyle (born 1 April 1961) is a Scottish singer, who rose to fame after appearing as a contestant on the third series of Britain's Got Talent, singing "I Dreamed a Dream" from . [SEP] Her debut studio album, I Dreamed a Dream, was released in November 2009 and became the UK's best-selling debut album of all time, beating the previous record held by Spirit by Leona Lewis.		
Sentence: Susan Magdalane Boyle (born 1 April 1961) is a Scottish singer who came to international attention when she appeared as a contestant on the TV programme Britain's Got Talent on 11 April 2009, singing "I Dreamed a Dream" from .		**Sentence**: Susan Magdalane Boyle (born 1 April 1961) is a Scottish singer, who rose to fame after appearing as a contestant on the third series of Britain's Got Talent, singing "I Dreamed a Dream" from .		
Susan Magdalane Boyle (born 1 April 1961) is a Scottish singer, who came rose to fame after appearing international attention when she appeared as a contestant on the TV programme third series of Britain's Got Talent, on 11 April 2009, singing "I Dreamed a Dream" from .				
Diff has new info?			Yes ▾	
Question	**Answer (NP)**	**Mark QA**	**Edited Question (if any)**	
What is Susan Boyle's real name?	Susan Magdalane Boyle	Edit ▾		
When was Susan Boyle born?	1 April 1961	Reject ▾		
Who is Susan Boyle?	a Scottish singer	Skip ▾		
In which series of Britain's Got Talent did Susan Boyle compete?	the third series	Accept ▾		
What show did Susan Boyle appear on?	Britain's Got Talent	Reject ▾		
What is the name of the song that Susan Boyle sang on Britain's Got Talent?	a Dream	Skip ▾		

Figure 5: Interface for the first phase of annotations, where an annotator chooses one of the five options: Accept/Reject/Edit/Context/Skip. Each example is annotated by two annotators. If both agree, the example is accepted as is or goes to a third annotator for editing. If one of the annotators skips, the third annotator makes the final decision.

DIFF 78: Susan Boyle (Id: 22378444_902516498_0)					
Base		**Target**			
Context: 2018. [SEP] Susan Magdalane Boyle (born 1 April 1961) is a Scottish singer who came to international attention when she appeared as a contestant on the TV programme Britain's Got Talent on 11 April 2009, singing "I Dreamed a Dream" from . [SEP] Her first album, I Dreamed a Dream, was released in November 2009 and became the UK's best-selling debut album of all time, beating the previous record held by Spirit by Leona Lewis.		**Context:** 2019. [SEP] Susan Magdalane Boyle (born 1 April 1961) is a Scottish singer, who rose to fame after appearing as a contestant on the third series of Britain's Got Talent, singing "I Dreamed a Dream" from . [SEP] Her debut studio album, I Dreamed a Dream, was released in November 2009 and became the UK's best-selling debut album of all time, beating the previous record held by Spirit by Leona Lewis.			
Sentence: Susan Magdalane Boyle (born 1 April 1961) is a Scottish singer who came to international attention when she appeared as a contestant on the TV programme Britain's Got Talent on 11 April 2009, singing "I Dreamed a Dream" from .		**Sentence:** Susan Magdalane Boyle (born 1 April 1961) is a Scottish singer, who rose to fame after appearing as a contestant on the third series of Britain's Got Talent, singing "I Dreamed a Dream" from .			
Susan Magdalane Boyle (born 1 April 1961) is a Scottish singer, who came rose to fame after appearing international attention when she appeared as a contestant on the TV programme third series of Britain's Got Talent, on 11 April 2009, singing "I Dreamed a Dream" from .					
Question	**Answer (NP)**	**Phase 1**	**Mark QA**	**Edited Question (if any)**	
x	Susan Magdalane Boyle	Edit/Skip	Edit ▾	Which Scottish singer appeared on the third season of Britain's Got Talent?	
x	a Scottish singer	Reject/Skip	Reject ▾		
On which television show did Susan Boyle appear?	the third series of Britain's Got Talent	Accept/Skip	Accept ▾		

Figure 6: Interface for the second phase of annotations, where a third annotator will rephrase a question and/or decide on a disagreed-upon annotation. Here, the annotator writes a new question for the answer span given the *Edit* annotation, and decides to confirm the *Reject* and *Accept* annotations of the other two examples. Note that for *Edit* or *Reject* annotations, to avoid bias, we do not display the seed-question to the annotators and instead display a *x*.