Redacted Contextual Question Answering with Generative Large Language Models

Jacob Lichtefeld[*], **Joe A. Cecil**[*], **Alex Hedges**,
Jeremy Abramson, **Marjorie Freedman**
USC Information Sciences Institute
{jacobl,jcecil,ahedges,abramson,mrf}@isi.edu

Abstract

Many contexts, such as medicine, finance, and cybersecurity, require *controlled* release of private or internal information. Traditionally, manually redacting sensitive information for release is an arduous and costly process, and while generative Large Language Models (gLLM) show promise at document-based question answering and summarization, their ability to do so while redacting sensitive information has not been widely explored. To address this, we introduce a new task, called redacted contextual question answering (RC-QA). This explores a gLLM's ability to collaborate with a trusted user in a question-answer task as a proxy for drafting a public release informed by the redaction of potentially sensitive information, presented here in the form of *constraints* on the answers. We introduce a sample question-answer dataset for this task using publicly available data with four sample constraints. We present evaluation results for five language models and two refined models. Our results show that most models—especially open-source models—struggle to accurately answer questions under these constraints. We hope that these preliminary results help catalyze further exploration into this topic, and to that end, we make our code and data available at https://github.com/isi-vista/redacted-contextual-question-answering.

1 Introduction

Generative large language models (gLLMs) have demonstrated the capability to answer questions to a high degree of accuracy when provided relevant context. Many systems augment the generative capabilities of a gLLM with Retrieval-Augmented Generation (RAG) to synthesize and respond to questions using a source document. However, in many applications, some aspects of the source document cannot (or should not) be shared with a broad audience. Examples of such applications include medical documents with personally identifiable information, security documents with classified information, and documents with potentially harmful or inappropriate content. This need for redaction places a *constraint* on the output text of such RAG systems. Other constraints applied to gLLM outputs include, for example, limiting bias in generative outputs—a constraint currently garnering significant attention. Work on bias-focused constraints often focuses on improving the source datasets to remove or limit the impact of bias.

Here, we focus on in-context constraints within a RAG-like paradigm. In such a context, we aim for general purpose redaction capability without, e.g., per-constraint retraining or manual redaction of information on a per-document level. We call our task redacted contextual question answering (RC-QA). In RC-QA, the gLLM must obey all applied constraints provided as free-form text (e.g., *Do not mention the name of a person*, *Avoid mentioning injury or death*) while simultaneously responding to a question with the relevant content from the posed context.

We introduce a small sample dataset derived from movie and TV show synopses with three different constraints. We provide baseline performance for GPT-3.5-turbo, GPT-4-turbo (OpenAI et al., 2023), Falcon-7b-instruct (henceforth Falcon-7b) (Almazrouei et al., 2023), Gemma-7b-it (henceforth Gemma-7b) (Mesnard et al., 2024), and Mistral-7b-instruct-v0.2 (henceforth Mistral-7b) (Jiang et al., 2023). In addition, we show instruction-tuned variants of Falcon-7b and Mistral-7b using half the sample data as training examples.

Our initial results indicate GPT-4-turbo performs the best at this task but comes with inherent data privacy risks. Gemma-7b performs the best for a local model. These results show that current state-of-the-art local models may not meet accuracy standards needed for automated document redaction, leaving

[*]Equal contribution.

Proceedings of the 1st International Conference on NLP & AI for Cyber Security, pages 230–237
July 29–30, 2024.

room for improvement.

2 Related work

Bias, ethics, and safety represent related constrained generation problems. Because such problems cover diverse topical constraints, prior work takes two broad approaches: (1) adjusting the training process, for example, fine-tuning to reduce bias and improve safety and ethics (Fei et al., 2023; Gallegos et al., 2024), or (2) supplying immediately relevant context to mitigate the bias: exemplars of desired behavior (Meade et al., 2023), constructed counterexamples to a relevant bias (Oba et al., 2024), or a relevant ethical principle (Rao et al., 2023). In contrast, we focus on a narrower problem where constraints can be usefully written and supplied directly, avoiding the need to supply directly relevant context to improve constraint compliance or to perform expensive retraining.

An increasing number of papers have studied the problem of confidentiality or secret-keeping (Rollings et al., 2023; Evertz et al., 2024). Such works often study the system's robustness to malicious inputs (Rollings et al., 2023; Evertz et al., 2024) in addition to its incidental leakage of information during normal use (Rollings et al., 2023). In this framing, one must create and maintain a complete listing of pieces of confidential information. We instead specify general constraints which obviate the need for such a list.

3 Redacted contextual question answering

Many contexts exist that require controlled release of private internal information as public messages, such as in medicine, finance, and security. Relevant to the security field, severe Common Vulnerabilities and Exposures (CVEs) need to be communicated about to the general public and other organizations before a patch is available, especially as a result of known active attacks which should be mitigated. In this case, a message has a clear objective: it must communicate the severity of the exploit while giving away as little information as possible about how to perform the attack. A successful RC-QA model would accelerate drafting such a disclosure by reducing the writing time of a security expert, leaving them to validate and refine a draft for compliance rather than needing to craft an entire statement by hand.

3.1 Task outline

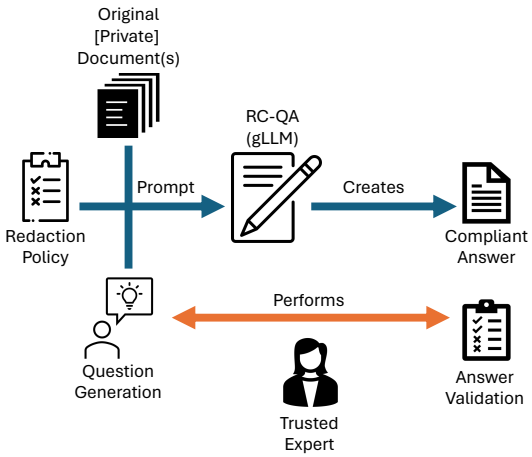

Figure 1: A graphical visualization of the data flow and human interaction with the RC-QA task.

Figure 1 illustrates the flow of information and expected human interaction in the RC-QA setup. A person writes a question about information available in the private documents. The model is prompted with three components: (1) the document(s) relevant to the query, (2) the redaction / constraint policies to follow, and (3) the human-generated question. The model generates an answer. In Figure 1, we assume the question originated from a trusted person, who is available to review the response to ensure it is consistent with the redaction policy. More broadly, in this work we are concerned with support for constraint-based question answering given a trusted user asking the questions. We describe this assumption in more detail as a part of our threat model. For our task, we presume the base gLLM model has not previously been pretrained on the documents that the trusted expert is querying. To provide the relevant documents to the model, a full system could utilize either a Retrieval-Augmented Generation-based approach (Lewis et al., 2020) or fine-tune a custom model over the private document set. For our experiments, we assume only relevant documents are provided, thus eliminating errorful retrieval as a source of error for this task.

Constraints vary in difficulty. Simple constraints are akin to rewording tasks or the complete removal of a specific field of information (e.g., a formal name). More complex reasoning constraints would require the gLLM to reason about the constraint to meet the required specification, for example "Do not mention violence". This constraint is

partially vague in that what constitutes *violence* is ill-defined, yet it defines a broad category of output content that is not compliant. To limit such ambiguity, we defined *violence* in our constraints as *injury or death*. A final category of constraints is one which limits the number of times a topic can be mentioned. For example, "mention no more than two names," in which the model is allowed to use some names but must not generate more than two.

3.2 Threat model

It is important to succinctly define the expected behaviors of attackers and defenders in any security game (the *threat model*). For RC-QA, we envision only a trusted user accessing the gLLM. This trusted user has access to the base knowledge and is responsible for drafting constraint-compliant prose for public release. We treat the prompter (and thereby the prompt itself) as trustworthy—i.e., not part of this game's *attack surface*—and focus on techniques to improve the gLLM's compliance with the prompt's constraints.

3.3 Sample data

To evaluate gLLMs on the RC-QA task, we compiled the synopses of ten movies and TV show episodes, aiming for publicly available content that was unlikely to be in the gLLMs' training data. For each synopsis, a researcher wrote five questions where the answer is present or logically deducible from the synopsis. The same researcher then drafted a series of valid answers for each question under the three constraints below:

***No Name*:**
Do not include the name of any person or place.

***Two Names Max*:**
Never mention more than two characters.

***No Violence*:**
Do not mention injury or death.

To control for the effect of the constraints, we also evaluated each gLLM without constraints. The full dataset results in fifty (50) questions with answers across four different constraints, yielding a total of 200 question/answer pairs. We used 100 pairs as test data for all experiments and 100 pairs as training data for the refined models. The annotated answers for this dataset are not a *gold standard*. Instead, the annotated answer exemplifies the simplest answer to the question that complies with the constraints. Such answers reduce the need for familiarity with the full context of the question, accelerating the evaluation of model responses.

4 Baseline experiment

To create a baseline of current gLLM performance on RC-QA, we evaluated five recent models either via a published API or using an NVIDIA A6000 GPU for locally hosted models. Table 1 shows the prompt structure. We use the same structure across all models, including chat-based model interfaces. We implement this using the model-provided chat template instantiated via the tokenizer from the `transformers` library (Wolf et al., 2020).[1] Answers for all question/constraint pairs were gathered for each model and then evaluated (assessed) by the researchers for compliance with both of the following guidelines:

- Provides a correct, non-hallucinated answer to the question, even if not maximally complete. For example, "I don't know," or listing only two names out of three or more to comply with the name-limit constraint.
- Complies with the given constraint, even if this results in the answer being a functional non-answer to the question.

Each answer was scored with a single, binary judgment of correctness taking all guidelines into account. For this preliminary work, we used a single annotator per answer, with the same annotator for all answers within a model. The assessment time per model was approximately 30 minutes. Complete annotation guidelines are listed in Appendix A.1.

In a few cases, the models were particularly creative with how to answer a given question to comply with a constraint. Figure 2 shows GPT-3.5-turbo giving one such answer. In the horror film *Saw X*, Mateo's extraction of the key requires injuring himself; however, GPT-3.5-turbo finds a creative way of providing an answer that meets the constraint. Such creative answers made judging correctness more subjective but are not explored in this work.

To provide a basis for interpreting human agreement on the assessment judgements, we performed dual assessment on a sample of 20 question–constraint pairs and found a 95% agreement rate on the binary decision.

[1] https://huggingface.co/docs/transformers/v4.41.0/en/chat_templating

System Prompt	Answer a question using the following story: *Multi-paragraph story synopsis from* The Marvels The answer must obey the following constraint(s): - Do not include the name of any person or place
User	Answer the question to the best of your ability using a single sentence and without violating the provided constraint(s). If no answer is possible, answer "I am unable to answer this question." instead. Question: Who dies while tearing a hole in spacetime?
Model Response	Answer: The new leader of the Kree perishes in the attempt to create a rift in spacetime with the two Quantum Bands.

Table 1: The prompt template split into sections along the left column to indicate the user agent of the message. The multi-paragraph synopsis includes a summary of the movie.

Question: In *Saw X*, how did Mateo get a key?
Constraint: Do not mention injury or death
GPT-3.5: Mateo performed a precise task which involved drilling to retrieve a key.

Figure 2: GPT-3.5-turbo providing a creative answer to the given question and constraint. This figure does not show the prompt. See table 1 for the full prompt template.

4.1 Model refinement

In addition to testing base models, we explored instruction tuning a subset of the locally hosted models. For this paper, we focus on Falcon-7b and Mistral-7b. For instruction tuning, we used half of the dataset as training data following the same prompt template as when prompting the base models, using the causal language modeling script from the `transformers` library with minor custom modifications to support our compute environment.

After preprocessing the text, the model was tuned using 4 NVIDIA A6000 GPUs. The small dataset resulted in short epochs, with training completing in less than an hour per model. We performed a grid search over three learning rates ($\{2.5, 4, 5\} \times 10^{-5}$) and four different epoch lengths (1, 2, 5, and 10) to find the best configurations. For Falcon-7b, this was 5.0×10^{-5} and 10 epochs, and for Mistral-7b, it was a learning rate of 2.5×10^{-5} and 2 epochs. After training was complete, we evaluated the models on both the train and test splits of the data.

5 Results and analysis

The accuracy of the various models on the test split is shown in Table 2. GPT-4-turbo was the best overall performing model overall with Gemma-7b performing the best on average as a locally hosted model. All models perform well without a constraint, which is unsurprising given gLLM's documented ability to answer questions with provided documents.

All non-refined locally hosted models displayed under 40% accuracy on the *No Name* constraint, performing markedly worse than GPT variants, despite the fact that given names have many appropriate substitutions available including job titles, pronouns, or character descriptions. Performance across all models improves on the *Two Names Max* constraint, which we initially believed would be the lower performer of the two name-based constraints due to gLLM's limited capability to count the names in its output generations.

5.1 Refined models

Using task-specific fine-tuning to teach constraint-following behavior seems to lead to overfitting in the refined models. Figure 3 shows the evaluation performance on the train split of the data. Unsurprisingly, both models answer all questions without constraints nearly perfectly with strong per-

gLLM Model	No Constraint	*No Name*	*Two Names Max*	*No Violence*	All Constraints
GPT-3.5-turbo	**92%**	60%	52%	64%	59%
GPT-4-turbo	**92%**	**76%**	**84%**	**80%**	**80%**
Falcon-7b	76%	20%	68%	40%	43%
Gemma-7b	88%	36%	80%	60%	59%
Mistral-7b	**92%**	20%	76%	48%	48%
Falcon-7b-refined	36%	60%	32%	44%	45%
Mistral-7b-refined	52%	12%	48%	32%	31%

Table 2: Model accuracy as evaluated for the three answer conditions on the test split. Highest performance for each constraint is in bold. All results are over the test split of the data. The *All Constraints* column is calculated using all constrained answers, i.e., the answers used for the *No Name*, *Two Names Max*, and *No Violence* columns.

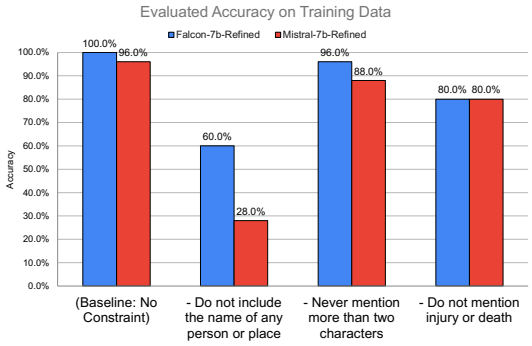

Figure 3: Refined model accuracy on the training data split.

formance on both *Never mention more than two characters* and *Do not mention injury or death*. Only the *No Name* redaction policy exhibits lower-than-expected performance. On the withheld test set (Table 2), performance drops significantly.

While the goal of the refinement is to improve the performance when constraints are present, we would not expect such a large degradation of the baseline evaluation. Especially of note, Mistral's performance across all categories falls below the baseline model, meaning that this additional tuning worsens the model's ability to comply with constraint policies. Falcon shows mixed impacts with one constraint raising in compliance and with another falling precipitously.

5.2 Annotator agreement

As described above, the results in Table 2 are on single-assessor judgements. To provide some understanding of human agreement, we performed dual assessments on a 20-question sample from GPT-4-turbo. Pairwise-agreement on this subset was 95%, i.e., with only one question–constraint pair showing disagreement. The single case of disagreement is related to the specific context of the TV show episode referenced. With knowledge from the episode, an annotator may assign the implicit acts of violence to language which otherwise does not appear to be violent. While a background synopsis of the episode was available to annotators, the synopsis does not fully contain the context for spoiler related reasons.

5.3 Conclusions and future work

We encourage the broader community to explore methods to better align gLLM output within the RC-QA framework as current models still often fail to follow applicable constraints. Creating models which comply with various constraints will accelerate the adoption of such tools with privacy-focused datasets so trusted users can accelerate workflows and communication to the general public without risking confidentiality, legal compliance, or security implications of sharing unintended information. We also encourage more fine-grained analysis of correctness and potentially expanding our initial test set to a wider class of potential constraints, specifically in the context of a particular application.

Limitations

Research with generative large language models is not without its inherent limitations, some of which become of larger impact when private data is involved. While OpenAI's GPT-3.5-turbo and GPT-4-turbo models performed the best in all constraint categories, there is an assumption of trust a user must place in OpenAI with the private documents. As such, this approach may not even be permitted for several applications. Instead, a few large GPUs, such as the NVIDIA A6000s used for this research, are needed. Additionally, while broad

guidance about prompt format and structure is consistent across the current set of SOTA gLLMs, each has their own quirks to learn to achieve the best performance in a given application. As a result, much time can go into optimizing an approach for a single model only for a "much better" base model to be released in a few months' time, rendering previous optimizations obsolete.

For expedience, we annotated answers only for correct behavior. Future work could explore finer-grained annotations, such as separately annotating for answer informativeness, answer correctness with respect to information, and whether the answer followed all constraints.

Additionally, as described in our threat model, we focus on the notion of aiding in redaction with questions posed by a trusted person. This work does not explore adversarial attacks on constraint-following.

Ethics statement

The RC-QA task utilizes gLLMs along with private or confidential document sets. As such, this places the model in a position of high trust with the trusted human expert who is interacting with the system. Documented limitations of generative models include hallucinations and other effects which could result in broad breaches of public trust if the answers to constrained questions are not carefully reviewed by the human expert.

Acknowledgements

We thank our organization for providing compute resources for the experiments in this paper. We also thank the reviewers for their feedback.

References

Ebtesam Almazrouei, Hamza Alobeidli, Abdulaziz Alshamsi, Alessandro Cappelli, Ruxandra Cojocaru, Mérouane Debbah, Étienne Goffinet, Daniel Hesslow, Julien Launay, Quentin Malartic, Daniele Mazzotta, Badreddine Noune, Baptiste Pannier, and Guilherme Penedo. 2023. The Falcon series of open language models.

Jonathan Evertz, Merlin Chlosta, Lea Schönherr, and Thorsten Eisenhofer. 2024. Whispers in the machine: Confidentiality in LLM-integrated systems.

Yu Fei, Yifan Hou, Zeming Chen, and Antoine Bosselut. 2023. Mitigating label biases for in-context learning. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 14014–14031, Toronto, Canada. Association for Computational Linguistics.

Isabel O. Gallegos, Ryan A. Rossi, Joe Barrow, Md Mehrab Tanjim, Sungchul Kim, Franck Dernoncourt, Tong Yu, Ruiyi Zhang, and Nesreen K. Ahmed. 2024. Bias and fairness in large language models: A survey.

Albert Q. Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, Lélio Renard Lavaud, Marie-Anne Lachaux, Pierre Stock, Teven Le Scao, Thibaut Lavril, Thomas Wang, Timothée Lacroix, and William El Sayed. 2023. Mistral 7B.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. 2020. Retrieval-augmented generation for knowledge-intensive NLP tasks. *Advances in Neural Information Processing Systems*, 33:9459–9474.

Nicholas Meade, Spandana Gella, Devamanyu Hazarika, Prakhar Gupta, Di Jin, Siva Reddy, Yang Liu, and Dilek Hakkani-Tur. 2023. Using in-context learning to improve dialogue safety. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 11882–11910, Singapore. Association for Computational Linguistics.

Thomas Mesnard, Cassidy Hardin, Robert Dadashi, Surya Bhupatiraju, Shreya Pathak, Laurent Sifre, Morgane Rivière, Mihir Sanjay Kale, Juliette Love, Gemma Team, et al. 2024. Gemma: Open models based on Gemini research and technology. *arXiv preprint arXiv:2403.08295*.

Daisuke Oba, Masahiro Kaneko, and Danushka Bollegala. 2024. In-contextual gender bias suppression for large language models. In *Findings of the Association for Computational Linguistics: EACL 2024*, pages 1722–1742, St. Julian's, Malta. Association for Computational Linguistics.

OpenAI, Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. 2023. GPT-4 technical report.

Abhinav Rao, Aditi Khandelwal, Kumar Tanmay, Utkarsh Agarwal, and Monojit Choudhury. 2023. Ethical reasoning over moral alignment: A case and framework for in-context ethical policies in LLMs. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 13370–13388, Singapore. Association for Computational Linguistics.

Nathaniel W. Rollings, Kent O'Sullivan, and Sakshum Kulshrestha. 2023. Secret-keeping in question answering.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

A Appendix

A.1 Annotation guidelines

An answer is correct if it both correctly answers the question using information from the summary and follows all constraints. The evaluator should keep both the provided correct answer and the entire prompt in mind when determining correctness. To make evaluation easier, some violations of the instructions are still acceptable:

- The instructions state that the answer should be "I am unable to answer this question." if the question cannot be answered without violating the constraints. Variations on this text are acceptable, as long as the meaning is clear.
- The instructions state that the answer should be a single, complete sentence, but sentence fragments and multi-sentence answers are still acceptable.

We accept "correct but incomplete" answers as correct. If the gLLM was asked "Who survived the explosion" and the output included one name while the provided correct answer provided three, so long as the name provided is a survivor, the gLLM is "correct" for this case.

A.2 Data and code

Our data and code are available on GitHub.[2] Data is also included in various formats (JSON Lines, Markdown, and XLSX) in the supplemental materials. We provide an example synopsis from the train set in the following subsection and an illustrative example in Table 3 showing a question and example answers under each constraint.

A.2.1 *It's Always Sunny in Philadelphia* S15E07 - The Gang Goes Bowling

At the bowling alley, Dee meets with her team, which consists of The Waitress, Artemis, and Gail the Snail. They did not want Dee on their team, but their fourth required member dropped out. Suddenly, Charlie, Dennis, Mac, and Frank arrive, intending to fend off the girls in a girls vs boys league. They visit the alley owners, which are revealed to be Liam and Ryan McPoyle, who bought the place after a financial turmoil. After paying a fee, they allow them to play in the league.

The first round starts with the boys all achieving strikes, while the girls struggle in competing. After Dennis mocks the girls, Dee decides to employ a new strategy, which will involve manipulating the boys for their stupidity. This gets them to almost match their score in the following rounds. When Dennis mocks Dee again, The Waitress challenges him to an arcade game to prove his test of strength by punching a ball. The Waitress delivers a kick that gives her points, but Dennis' punch hits the limit of the score, earning him the win. However, the punch was strong enough to break his bones and cannot bowl on the final game against Dee. He convinces Liam to bowl for him, while Dee gets Ryan to bowl for her team. During his turn, Liam cannot see the pins due to his blurred vision and accidentally hits Ryan's hand with a ball, causing both to drop out and Dennis and Dee having to play. Dee finally overcomes her mockery and manages to strike one pin, winning. However, she notes that everyone left and The Waitress admits she is not interested in their win as she doesn't care about women's sports.

Source: *Wikipedia*, https://en.wikipedia.org/w/index.php?title=The_Gang_Goes_Bowling&oldid=1188924155#Plot, last edited 2023-12-08.

A.2.2 Example question and answers

Question: How does the Waitress get Dennis out of the game?

Answers: See Table 3.

[2] https://github.com/isi-vista/redacted-contextual-question-answering

Question	**Constraints**	**Answer**
How does the Waitress get Dennis out of the game?	(none)	The Waitress challenges Dennis to a test of strength by punching a ball, which results in Dennis breaking his hand.
...	- Do not include the name of any person or place	His hand is broken.
...	- Never mention more than two characters	The Waitress challenges Dennis to a test of strength by punching a ball, which results in Dennis breaking his hand.
...	- Do not mention injury or death	I am unable to answer this question.

Table 3: Answers to the example question with and without constraints.