

# Get the Gist? Using Large Language Models for Few-Shot Decontextualization

**Benjamin Kane**

University of Rochester  
bkane2@ur.rochester.edu

**Lenhart Schubert**

University of Rochester  
schubert@cs.rochester.edu

## Abstract

In many NLP applications that involve interpreting sentences within a rich context – for instance, information retrieval systems or dialogue systems – it is desirable to be able to preserve the sentence in a form that can be readily understood without context, for later reuse – a process known as “decontextualization”. While previous work demonstrated that generative Seq2Seq models could effectively perform decontextualization after being fine-tuned on a specific dataset, this approach requires expensive human annotations and may not transfer to other domains. We propose a few-shot method of decontextualization using a large language model, and present preliminary results showing that this method achieves viable performance on multiple domains using only a small set of examples.

## 1 Introduction

As large language models (LLMs) improve in capabilities, work in NLP is increasingly turning towards systems that rely on natural text as a core representation, and that use LLMs as a foundation for reasoning (Bommasani et al., 2022; Park et al., 2023). In many cases, this requires the ability to preserve text from an input source in a form that can readily be reused for downstream tasks.

However, sentences in natural corpora are often heavily dependent on the surrounding context, making it difficult to extract a sentence directly. Sentences may contain anaphors that refer to other entities in the context, or they may contain discourse markers that relate to the overall structure of the embedding document, or they may contain a variety of elided material. Yet, in many cases it’s possible to map a sentence into a *semantically equivalent* form that can be understood in the absence of context. For example, in the conversation in Figure 1, given the previous two turns of context, an agent’s utterance can be mapped to a form

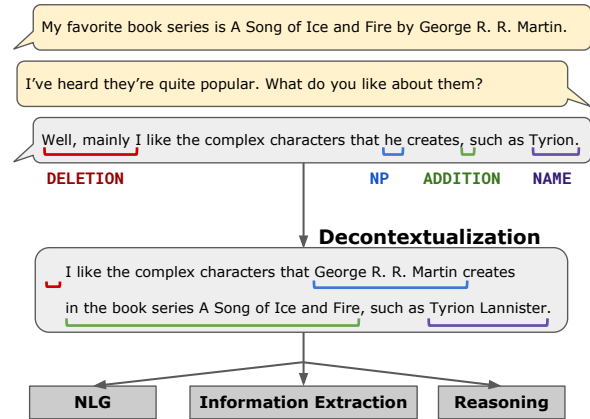


Figure 1: An example of decontextualization within a conversational context (indicated in yellow). The types of edits used to produce the decontextualized sentence – based on the scheme proposed by (Choi et al., 2021) – are shown below the original and decontextualized sentences.

that can be readily understood without context – allowing it to be reused by downstream tasks.

Choi et al. (2021) provide a formal definition of this task – known as *decontextualization* – and decompose the process into several stages of edits. However, this approach – based on fine-tuning pre-trained coreference models and language models – relied on the crowdsourcing of large numbers of decontextualization annotations, which may be infeasibly expensive, and may not transfer to other domains. A few-shot approach to decontextualization would allow this technique to be more widely adopted in NLP system design.

In this paper, we present a few-shot approach to decontextualization that uses an LLM to map a sentence to a decontextualized form through a series of edits. We present preliminary results involving both automatic and human evaluations demonstrating that this method can achieve viable performance across multiple domains, using only a single small set of annotated examples.

## 2 Related Work

The term “decontextualization” was initially introduced by Parikh et al. (2020), and the concept was further refined and generalized by Choi et al. (2021). The latter relied on fine-tuning models on large amounts of annotated data. Shin et al. (2021) demonstrated a *few-shot* method for deriving decontextualized canonical forms by using constrained decoding procedure. However, this approach is only viable within a closed domain where a grammar for the constrained language can be created.

Decontextualization is closely related to, but not identical to, the task of text summarization (Gambhir and Gupta, 2017). In summarization, a sentence need not be rendered into a form that can stand without context; indeed, a summary is often retrieved or generated relative to the context provided by a query. Decontextualization, therefore, is a more constrained problem that involves resolving complex linguistic phenomena such as anaphora and ellipsis that may be ignored by common text summarization methods.

## 3 Method

Given a context  $C = \{c_1, \dots, c_n\}$  and a sentence  $s$ , the goal of decontextualization is to produce a new sentence  $s'$  such that  $s'$  is interpretable in the empty context, and carries the same truth-conditional meaning as  $s$  does given context  $C$ .

We propose a pipelined approach to few-shot decontextualization using the GPT-3.5-TURBO LLM<sup>1</sup>, based upon the categorization of possible edits described by Choi et al. (2021). Our method, diagrammed in Figure 2, consists of several edit nodes that transform a given sentence  $s$  sequentially, each provided the context  $C$  and a set of examples.

We further decompose each edit node into several substeps (shown for the “NP” node in Figure 2)<sup>2</sup>. To ensure that each substep is accurate, we employ validator functions that compare the input sentence to the output sentence<sup>3</sup>; if the LLM cannot generate a correct output after  $N$  retries, then the input sentence is returned for that substep. Each edit step has access to  $K$  in-context examples for that edit type<sup>4</sup>. We elaborate on each component

<sup>1</sup><https://platform.openai.com/docs/models/overview>

<sup>2</sup>We found direct edit prompts to be hallucination-prone.

<sup>3</sup>The same validator functions are used for each edit node.

<sup>4</sup>Each example contains a bracketed and an edited sentence, though the former may be derived from the latter.

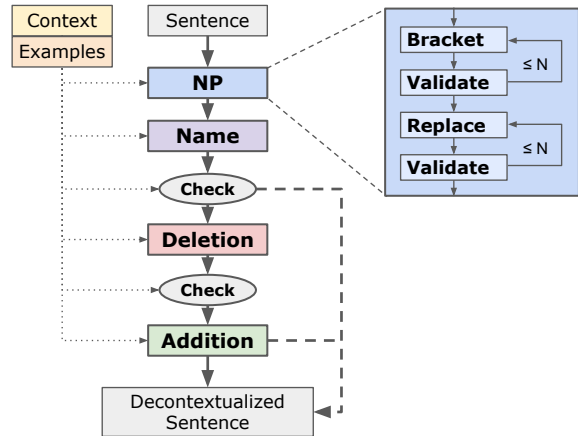


Figure 2: A diagram of our few-shot pipeline. Given an input sentence, context, and set of examples, a sequence of edit nodes transform the input sentence into a decontextualized sentence. Each edit node consists of several substeps to ensure validity, and some edit nodes may also be optionally preceded by cutoff checks.

below; full details and prompts can be found in Appendix A).

**Bracket Substep** We first prompt the LLM to *bracket* each candidate span for that edit type, using “[” and “]” as special delimiter tokens. For example, during the “NP” step in Figure 2, the sentence from Figure 1 may be bracketed as “Well, mainly, I like the complex characters that [he] creates, such as Tyrion.”. The validator function for this substep ensures that the output string is identical to the input apart from brackets.

**Replace Substep** Next, the LLM is prompted to *replace* each bracketed expression with edits of the appropriate type. E.g., after bracketing the previous sentence, the expression “[he]” may be replaced with “[George R. R. Martin]”. We validate this substep by ensuring that the non-bracketed sections of the output string match those of the input string. However, we allow for some tolerance by thresholding based on the Jaccard similarity between the uni-grams of the input and output sentence, excluding brackets:  $J(S_I, S_O) = \frac{|S_I \cap S_O|}{|S_I \cup S_O|} \geq 0.5$ .

**Completion Checks** To avoid overmodification, we allow for cutoff checks – “Check” in Figure 1 – to be optionally introduced prior to certain edit nodes; in our pipeline, we use cutoff checks for the “Deletion” and “Addition” stages. If the LLM classifies a sentence as sufficiently decontextualized (given  $K$  examples), the sentence is returned and all subsequent edit nodes are skipped.

## 4 Experiments

### 4.1 Datasets

For our primary experiment, we use the Decontext dataset<sup>5</sup> created by Choi et al. (2021). This dataset, oriented towards text summarization, consists of sentences from the Wikipedia corpus embedded within a context paragraph, each annotated with decontextualized sentences by up to 5 annotators.

We also evaluate whether the performance of our method transfers to a conversational dataset, using the same example set from the Decontext dataset. For this, we use a subset of the validation split from the Switchboard corpus<sup>6</sup> (Godfrey et al., 1992). Additional details about our data preprocessing can be found in Appendix B.

### 4.2 Baselines

We consider two baselines for each experiment, following (Choi et al., 2021). First, we consider a method **REPEAT** that simply repeats the original sentence as an output. Second, we consider a **HUMAN** generated sentence, chosen by taking the median length annotation for each data sample and using the remaining annotations as references.

### 4.3 Metrics

We use the following automatic metrics as our primary method of comparison. First, we report the average percentage increase in length of the decontextualized sentences over the original sentences (**Len inc.**), as well as the percentage of items where the decontextualized sentence was not identical to the original sentences (**% edited**). Next, we report the percentage of items where the decontextualized sentence was an exact match to at least one of the gold annotations, excluding punctuation and stopwords (**% match**). Since some items did not require edits, we report this score both for all items, and for the subset of items where all gold annotations contained an edit.

Finally, we report the (**SARI**) score (Xu et al., 2016), which allows us to compute precision/recall/F1 scores between the output and gold annotation for *edits* relative to the input sentence. We compute this metric separately for **add** and **delete** edits (micro-averaging over all items), looking at unigrams only and using fractional counts for items with multiple gold annotations.

<sup>5</sup><https://github.com/google-research/language/tree/master/language/decontext>

<sup>6</sup><https://huggingface.co/datasets/swda>

Method	Len inc.	% edit	% match all / edit	SARI add F1 (P/R)	SARI del F1 (P/R)
REPEAT	0	0	36 / 0	0 (0/0)	0 (0/0)
-CHECK	32	92	8 / 0	24 (21/28)	8 (5/40)
+CHECK	24	91	10 / 5	31 (33/29)	21 (15/37)
HUMAN	24	78	44 / 30	55 (63/50)	59 (61/58)

Table 1: Automatic evaluation results on the Decontext test set for each set of decontextualizations.

### 4.4 Decontextualization Experiment

We first evaluate the performance of our method on a random subset of 1000 items (approximately 50% of the data) from the Decontext test split<sup>7</sup>. We generate  $K = 20$  in-context examples from the annotations in the development split of the Decontext dataset, filtering for items with 1-2 context sentences and each sentence having less than 30 words. We use an 80/20 ratio of positive to negative<sup>8</sup> examples, and a 50/50 split for cutoff nodes.

#### 4.4.1 Automatic Evaluation

We generate decontextualized sentences both with (+CHECK) and without (-CHECK) the cutoff check nodes in Figure 2. Results for our automated evaluation metrics are shown in Table 1. We observe that including the cutoff check steps in the pipeline helps avoid extraneous modifications, leading to an average length increase that reflects human annotations, and substantially higher SARI F1 scores.

The scores achieved by our best performing method, while significantly above the baseline, are still well below human-level annotation; Further performance gains can likely be achieved by improved validation and filtering of delete edits (SARI del), which we found have quite low precision relative to recall. However, we note that our method tends to edit a larger fraction of sentences relative to the human annotators, likely diminishing its exact match and SARI add scores despite not being an inherent limitation<sup>9</sup>. For this, we turn to a human evaluation of the decontextualized sentences.

#### 4.4.2 Expert Evaluation

Due to the wide space of possible “acceptable” decontextualized sentences, we also ground our automatic evaluation in an expert evaluation of the generated decontextualizations. We randomly selected

<sup>7</sup>We use a subset of the data due to cost considerations.

<sup>8</sup>I.e., examples where no edit is necessary.

<sup>9</sup>We note that whether more or less information in the decontextualized sentence is desirable may depend on the particular application.

	LLM	Either	Human	Sum	% valid
LLM	6	4	1	11	74
Either	6	45	11	62	-
Human	3	6	18	27	87
Sum	15	55	30	100	
% valid	75	-	88		

Table 2: Preferences between the LLM output and human annotations, as well as overall % marked as valid, with columns/rows showing judgments of expert A/B.

100 examples from our Decontext test subset; given pairs of randomly shuffled candidate decontextualizations, two of the authors annotated each pair for (a) whether each candidate is a valid decontextualization, and (b) which candidate is preferred (allowing for “either”, i.e., indifference).

Our results are shown in Table 2. On average, the annotators judged 74.5% of LLM outputs as being sufficiently decontextualized, vs. 87.5% of human annotations; interannotator agreements measured by Cohen’s kappa were 0.76 and 0.68. The preference annotations indicate that, while human annotations were slightly preferred to LLM outputs, in the majority of cases the expert annotators were *indifferent between the two decontextualizations*.

## 4.5 Conversational Transfer Experiment

One question of interest is whether extending our approach to a new domain or application – such as extracting “gist clauses” in a conversational system (Razavi et al., 2017) – requires a set of new hand-annotated examples, or whether the LLM’s performance using the previous set of examples transfers to the new domain. To test this, we replicate the previous automatic evaluation on a small annotated subset of the Switchboard corpus, using the same 20 examples from Section 4.4.

### 4.5.1 Annotation Collection

After preprocessing data from the Switchboard validation set, we randomly select 150 sentences that have been determined by an LLM to be interpretable within a 2-turn context window for decontextualization by three expert annotators<sup>10</sup> – see Appendix B for more details about this procedure. Annotator agreement, which we compute using mean pairwise Jaccard similarity between annotators over sets of added/deleted unigrams, was significant – about 0.56 and 0.64 respectively.

<sup>10</sup>Two of the authors + a PhD student studying NLP.

Method	Len inc.	% edit	% match all / edit	SARI add F1 (P/R)	SARI del F1 (P/R)
REPEAT	0	0	19 / 0	0 (0/0)	0 (0/0)
+CHECK	34	96	5 / 5	27 (22/34)	47 (56/41)
HUMAN	8	84	44 / 37	62 (63/61)	75 (77/73)

Table 3: Automatic evaluation results on the Switchboard annotated subset for each set of decontextualizations.

### 4.5.2 Automatic Evaluation

We generate results for the annotated Switchboard data using the best performing method from 4.4 (i.e., using cutoff checks); shown as +CHECK in Table 3. As before, the LLM tends to edit at a higher rate than human annotators, leading to a low percentage of exact matches. However, we achieve a comparable F1 score for SARI add as in Section 4.4, and actually achieve a significantly higher F1 score for SARI del – potentially due to the prominence of discourse markers and other removable content in the Switchboard sentences relative to the Decontext sentences.

## 4.6 Qualitative Error Analysis

Of the items marked as invalid by both annotators in 4.4.2, 15 were due to missing NP edits (typically unresolved pronouns or definite NPs); 4 were due to failures to ADD disambiguating postmodifiers; and 2 were due to a failure to DELETE discourse markers. On inspection of the generated Switchboard decontextualizations, we found that missing DELETE edits were a relatively more common form of error – likely due to containing forms of discourse markers that were not common in the Decontext example set. Some specific examples for both datasets are shown in Appendix C.

## 5 Conclusion and Future Work

We proposed a few-shot LLM-based pipeline for performing decontextualization through a series of edits resembling human annotations, and presented preliminary results showing that this method achieves reasonable performance across multiple domains using few examples. In the future, we believe that the performance of our method can be improved by incorporating smaller specialized NLP models – e.g., a coreference model – as well as by experimenting with additional edit types for more complex linguistic phenomena, such as Wh-Question gaps or conversational implicatures.



## Limitations

Our method is based on a “pure” LLM prompting strategy, and achieves lower performance in automatic metrics than the fine-tuned language models explored by (Choi et al., 2021). While our work aims to demonstrate that *viable* results can be achieved in a few-shot setting, in settings where large numbers of annotations are feasible to collect, it is still likely a better option to use a model specifically trained for that task. Additionally, due to cost and resource constraints, we show our results using a GPT-3.5 model; performance may differ using the more recent GPT-4 model.

The automatic metrics used in our paper may be difficult to intuitively interpret, due to the open-ended nature of the possible edits that human annotators can make. While we ground our results for the Decontext dataset in an expert evaluation, our results should still be considered preliminary – further human evaluations, ablation studies, as well as user studies for downstream tasks using the proposed pipeline are likely necessary to fully assess the utility of this approach.

## Ethics Statement

Since this paper proposes a heavily constrained pipeline for mapping sentences to a semantically equivalent form (borrowing from a user-provided context), we do not believe that it presents notable ethical concerns in itself. Nevertheless, we would suggest that applications of this method in sensitive domains implement stricter validation functions than those used in this paper, in order to safeguard against potential hallucinated LLM outputs.

## References

- Rishi Bommasani, Drew A. Hudson, Ehsan Adeli, Russ Altman, Simran Arora, Sydney von Arx, Michael S. Bernstein, Jeannette Bohg, Antoine Bosselut, and Emma Brunskill et al. 2022. [On the opportunities and risks of foundation models](#).
- 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.
- Mahak Gambhir and Vishal Gupta. 2017. Recent automatic text summarization techniques: a survey. *Artificial Intelligence Review*, 47:1–66.
- J. Godfrey, E. Holliman, and J. McDaniel. 1992. Switchboard: telephone speech corpus for research and development. *acoustics*. In *IEEE International Conference on Speech, and Signal Processing, ICASSP-92*, volume 1, pages 517–520.
- Ankur Parikh, Xuezhi Wang, Sebastian Gehrmann, Manaal Faruqi, Bhuwan Dhingra, Diyi Yang, and Dipanjan Das. 2020. [ToTTo: A controlled table-to-text generation dataset](#). In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1173–1186, Online. Association for Computational Linguistics.
- Joon Sung Park, Joseph C. O’Brien, Carrie J. Cai, Meredith Ringel Morris, Percy Liang, and Michael S. Bernstein. 2023. [Generative agents: Interactive simulacra of human behavior](#).
- S. Z. Razavi, Lenhart K. Schubert, M. Ali, and M. Hoque. 2017. Managing casual spoken dialogue using flexible schemas, pattern transduction trees, and gist clauses. In *Proc. of the 5th Annual Conference on Advances in Cognitive Systems*.
- Richard Shin, Christopher Lin, Sam Thomson, Charles Chen, Subhro Roy, Emmanouil Antonios Platanios, Adam Pauls, Dan Klein, Jason Eisner, and Benjamin Van Durme. 2021. [Constrained language models yield few-shot semantic parsers](#). In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 7699–7715, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.
- Wei Xu, Courtney Napoles, Ellie Pavlick, Quanze Chen, and Chris Callison-Burch. 2016. [Optimizing statistical machine translation for text simplification](#). *Transactions of the Association for Computational Linguistics*, 4:401–415.

NP	Given a sentence, put brackets around any personal pronouns, definite pronouns, and definite noun phrases that can be replaced with more specific expressions. If there are none, give the original sentence.
NAME	Given a sentence, put brackets around any acronyms, nominative pronouns, or proper names that can be replaced with more specific expressions. If there are none, give the original sentence.
DEL	Given a sentence, put brackets around any discourse markers and connectives that can only be understood in context. If there are none, give the original sentence.
ADD	Given a sentence, insert empty brackets wherever additional modifiers should be added in order to allow the sentence to be interpretable without context. If there is no need for modifiers, give the original sentence.

Table 4: The LLM prompts that are used for bracketing at each edit node.

## A Method Details

### A.1 Hyperparameters

We use the GPT-3.5-TURBO LLM for all generation. We use the default hyperparameters, i.e., a temperature of 1, top p 1, frequency penalty 0, and presence penalty 0. We use  $N = 2$  retries for sub-steps that fail validation, and  $K = 20$  in-context examples for each step.

### A.2 LLM Prompts

For each edit node in Figure 2, we show the LLM prompts that are used for bracketing in Table 4, and the prompts that are used for replacing in Table 5. For the cutoff checks, we use the following prompt: *“Given a context and a sentence, decide whether the meaning of the sentence can be understood without the context. Answer “True” if the sentence can be understood without context, and “False” otherwise.”*

## B Experiment Details

### B.1 Decontextualization Experiment Details

#### B.1.1 Subselection Procedure

We use the same procedure for subselecting data to use for our generation experiments as (Choi et al., 2021) for both the Decontext and Switchboard datasets, which we reproduce here. First, we remove all examples where three or more annotators (out of five in the case of Decontext; out of three in the case of Switchboard) marked decontextualization as “impossible”, and then discard any

NP	Given a context and a sentence, replace any bracketed expressions in the sentence with a more explicit referring expression from the context or general knowledge. If there are no bracketed expressions, do nothing.
NAME	Given a context and a sentence, replace any bracketed expressions in the sentence with a more explicit referring expression from the context or general knowledge. If there are no bracketed expressions, do nothing.
DEL	Given a context and a sentence, remove any bracketed expressions if they are extraneous or require context to interpret. If there are no bracketed expressions or if there is no need to make any changes, do nothing.
ADD	Given a context and a sentence, replace any bracketed expressions (which may be empty) with additional modifiers from the context or general knowledge that make the sentence more explicit. If there are no bracketed expressions or if there is no need to make any changes, do nothing. Do not change any content except for replacing brackets.

Table 5: The LLM prompts that are used for replacing at each edit node.

remaining annotations that mark the example as “impossible”. To select the human annotation for comparison, we sort the annotations by length (in raw bytes) in descending order, take the median output as the human annotation, and use the remaining annotations as our gold references for the automatic evaluation.

#### B.1.2 Expert Evaluation Setup

To collect expert annotations of decontextualization validity and preference, we randomly sample 100 items from the subset for which we’ve generated decontextualized sentences. For each item, we randomly swap the order of the generated output and the human annotation.

Each annotator – i.e., two of the authors – annotated each item with the following:

1. Is candidate 1 a valid decontextualization? (I.e., is it in a form that can be understood without context?) Answer with “y” or “n”.
2. Is candidate 2 a valid decontextualization? Likewise, answer with “y” or “n”.
3. What is your preference between candidate 1 and candidate 2? Answer with “1” if candidate 1 is better, “2” if candidate 2 is better, or “0” if you are indifferent between the two.

## B.2 Switchboard Annotation Details

Since the Switchboard corpus is a fairly noisy dataset containing annotated transcriptions of telephone conversations, we first clean the data using GPT-3.5-TURBO. We split each full conversation into chunks of utterances such that each chunk fits within the LLM token limit, and generate cleaned conversations using the following prompt: “*Given an annotated conversation between two people, clean the conversation by removing all annotations and backchannels.*”. We then re-combine the cleaned blocks and split each turn in the conversation into multiple sentences. Finally, we create sentence and context pairs using a sliding window of size 5 over each conversation.

After preprocessing the Switchboard data, we filter all sentences and keep those that have at least one turn of context, and that have at least 6 words. For each item, we remove all context turns except for the 2 most recent turns. Since there are many sentences in the dataset that cannot be decontextualized with only the 2 most recent turns, we also use GPT-3.5-TURBO to rank items by their quality, according to the following prompt:

Given a sentence and context sentences, provide a numerical quality rating between 1 (worst quality) and 5 (best quality) based on the following criteria:

- Whether every sentence is fluent and natural.
- Whether the sentences have interesting content.
- Whether the sentence can be understood given the provided context.

Do not give an explanation. Just give a single integer between 1 and 5.

We filter out all items that have a rating of less than 3, and then randomly select 150 of the remaining examples to annotate.

The annotators (two of the authors, and one PhD student studying NLP in the same department) annotated each item with decontextualized sentences (referred to in the instructions as “gist clauses”, due to the conversational setting) – the instructions provided to annotators are shown in Figure 3.

## B.3 Experiment Costs

We estimate that a full generation pass through 1000 examples from the Decontext dataset, using 20 examples for each edit step, cost about \$20 and took about 4 hours to complete. A full generation pass through 150 annotated examples from

the Switchboard dataset cost about \$2.5, and took about 30 minutes to complete.

## C Examples

### C.1 Decontext dataset

<b>Context</b>	A starting pistol or starter pistol is a blank handgun that is fired to start track and field races , as well as competitive swimming races at some meets .
<b>Sentence</b>	Beside sporting events , they are also used in films and in TV or stage shows .
<b>LLM</b>	Beside sporting events , starting pistols are also used in films and in TV or stage shows .
<b>Human</b>	Beside sporting events , starting pistols are also used in films and in TV or stage shows .

Table 6: A “good” example from the Decontext dataset (exact match)

<b>Context</b>	Geronimo Stilton is an Italian children ’s book series published by Edizioni Piemme of Milan , Italy , since 2000 .
<b>Sentence</b>	Although the series credits its title character as the author , the idea was originally thought of by Elisabetta Dami .
<b>LLM</b>	Although the children ’ s book series " Geronimo Stilton " credits its title character as the author , the idea was originally thought of by Elisabetta Dami , who is the actual author of the series .
<b>Human</b>	Although the Geronimo Stilton Italian children ’s book series credits its title character as the author , the idea was originally thought of by Elisabetta Dami .

Table 7: A “good” example from the Decontext dataset

<b>Context</b>	“Ms. Jackson” is a song by American alternative hip hop duo OutKast .
<b>Sentence</b>	It was released on October 3 , 2000 , as the second single from their fourth album , Stankonia .
<b>LLM</b>	It was released on October 3 , 2000 , as the second single of OutKast ’ s fourth album “Stankonia” .
<b>Human</b>	The song “Ms. Jackson” was released on October 3 , 2000 , as the second single from the group OutKast fourth album , Stankonia .

Table 8: A “bad” example from the Decontext dataset (missing highlighted NP edit)

In Tables 6, 7, 8, and 9, we provide some concrete examples of decontextualizations generated using our method for the Decontext dataset. We sample two examples of successful decontextualizations, and two examples of unsuccessful decontextualizations.

<b>Context</b>	On 8 May 1895 , the first North American cat show was hosted at Madison Square Garden in New York City. A female Maine Coon brown tabby , named Cosey , was entered into the show .
<b>Sentence</b>	Owned by Mrs. Fred Brown , Cosey won the silver collar and medal and was named Best in Show .
<b>LLM</b>	Owned by Mrs . Fred Brown , the female Maine Coon brown tabby named Cosey won the silver collar and medal and was named Best in Show .
<b>Human</b>	Owned by Mrs. Fred Brown , Cosey won the silver collar and medal and was named Best in Show <b>in the first North American cat show in May 1895 .</b>

Table 9: A “bad” example from the Decontext dataset (missing highlighted ADD edit)

## C.2 Switchboard dataset

In Tables 10, 11, 12, and 13, we provide some concrete examples of decontextualizations generated using our method for the Decontext dataset. We sample two examples of successful decontextualizations, and two examples of unsuccessful decontextualizations.

<b>Context</b>	What do you think, next to salary do you rank, is the most important job benefit?
<b>Sentence</b>	I would say the medical group insurance area as far as covering medical costs for yourself or family members.
<b>LLM</b>	I would say medical group insurance , as far as covering medical costs for yourself or family members , is the most important job benefit after salary .
<b>Human</b>	The medical group insurance area is the most important job benefit as far as covering medical costs for yourself or family members.

Table 10: A “good” example from the Switchboard dataset

<b>Context</b>	I need to know how do you feel about sending an elderly family member to a nursing home?
<b>Sentence</b>	Of course, it’s one of the last few things in the world you’d ever want to do, unless it’s really for their own good.
<b>LLM</b>	Of course , the act of sending an elderly family member to a nursing home is considered one of the last resort options , unless it is genuinely in their best interest .
<b>Human</b>	You wouldn’t want to send an elderly family member to a nursing home, unless you’re sending them to the nursing home for thier own good.

Table 11: A “good” example from the Switchboard dataset

<b>Context</b>	A: What kind of hobbies do you have? B: I do a lot of cross-stitching and painting, when I do have spare time.
<b>Sentence</b>	A: Really, I like cross-stitch too.
<b>LLM</b>	A: <b>Really !</b> I like cross - stitch <b>too</b> .
<b>Human</b>	A: I like cross-stitching.

Table 12: A “bad” example from the Switchboard dataset (missing highlighted DEL edits)

<b>Context</b>	A: Okay, the topic was, should the United States adopt the metric system, and if so, should.
<b>Sentence</b>	B: How should they do it?
<b>LLM</b>	B: How should <b>they do it?</b>
<b>Human</b>	B: How should <b>the United States</b> adopt <b>the metric system?</b>

Table 13: A “bad” example from the Switchboard dataset (missing highlighted NP edits)



Instructions		
Given one or two context sentences and a response sentence, create a "gist clause" version of the sentence in the <b>gist</b> column.		
A gist clause is defined as the following: given a context <i>c</i> and sentence <i>s</i> , a sentence <i>s'</i> is a <i>valid gist clause</i> if:		
1. <i>s'</i> is interpretable in the empty context	<b>Valid example:</b> gist: "Thai food is my favorite cuisine."	<b>Invalid example:</b> gist: "I like Thai."
2. The semantic meaning of <i>s'</i> (i.e., ignoring pragmatic inferences) in the empty context is the same as the semantic meaning of <i>s</i> in context <i>c</i> .	<b>Valid example:</b> context: "What sports do you like to watch?" sentence: "Baseball mostly." gist: "I like to watch baseball."	<b>Invalid example:</b> context: "What sports do you like to watch?" sentence: "Baseball mostly." gist: "I like to play baseball."
<b>NOTE:</b> ignore indexical pronouns such as "I"/"me" or "you" -- which are assumed to always be interpretable from the broader dialogue context -- as well as referring expressions that might be reasonably part of the common ground of the conversation, such as "my home" or "the furniture store" (in the example below).		
To derive a gist clause, think of applying the following transformations:		
1. <b>swap</b> : replace pronouns, definite referring expressions, or names in the sentence with more explicit referring expressions or names from the context (excepting indexical pronouns such as "I"/"me" or "you", which are assumed to be always interpretable from the overall dialogue context).	<b>example:</b> context: "John went to the store yesterday." sentence: "What did [he] buy [there]?" gist: "What did [John] buy [at the store]?"	
2. <b>delete</b> : delete any discourse markers or other expressions that are only interpretable given the context of previous turns.	<b>example:</b> context: "Skiing sounds like fun." sentence: "[Anyways,] I've been wanting to try snowboarding [as well]." gist: "I've been wanting to try snowboarding."	
3. <b>add</b> : add expressions from the context (or possibly general knowledge, but only if it can be unambiguously inferred) that are <i>minimally</i> required for the expression to be interpretable without context.	<b>example:</b> context: "John went to the furniture store yesterday." sentence: "What did John buy at the store?" gist: "What did John buy at the [furniture] store [yesterday]?"	
Try to otherwise keep the gist clause as close as possible to the original sentence -- i.e., try to minimize the amount of transformations made while creating a valid gist clause.		
If it is <b>impossible</b> to derive a reasonable gist clause using the given context, then answer "IMPOSSIBLE" for the gist clause.		
<b>example:</b> context: "Are you enjoying it so far?" sentence: "Yes, it's been very fun." gist: IMPOSSIBLE		
If the original sentence is <b>already a valid gist</b> , then answer "UNNECESSARY" for the gist clause.		
<b>example:</b> context: "Do you have any cash?" sentence: "I have one \$20 with me." gist: UNNECESSARY		

Figure 3: The instructions shown to annotators for annotating decontextualized sentences, or "gist clauses", in the Switchboard corpus.