arXiv:2205.09641v1 [cs.CL] 19 May 2022

SNAC: Coherence Error Detection for Narrative Summarization

Tanya Goyal[1] **Junyi Jessy Li**[2] **Greg Durrett**[1]
[1] Department of Computer Science [2] Department of Linguistics
The University of Texas at Austin
tanyagoyal@utexas.edu

Abstract

Progress in summarizing long texts is inhibited by the lack of appropriate evaluation frameworks. When a long summary must be produced to appropriately cover the facets of that text, that summary needs to present a coherent narrative to be understandable by a reader, but current automatic and human evaluation methods fail to identify gaps in coherence. In this work, we introduce SNAC, a narrative coherence evaluation framework rooted in fine-grained annotations for long summaries. We develop a taxonomy of coherence errors in generated narrative summaries and collect span-level annotations for 6.6k sentences across 150 book and movie screenplay summaries. Our work provides the first characterization of coherence errors generated by state-of-the-art summarization models and a protocol for eliciting coherence judgments from crowd annotators. Furthermore, we show that the collected annotations allow us to train a strong classifier for automatically localizing coherence errors in generated summaries as well as benchmarking past work in coherence modeling. Finally, our SNAC framework can support future work in long document summarization and coherence evaluation, including improved summarization modeling and post-hoc summary correction.[1]

1 Introduction

As pre-trained models for news summarization (Lewis et al., 2020; Zhang et al., 2020; Brown et al., 2020) have improved drastically in recent years, researchers have begun tackling increasingly challenging settings, particularly long-document summarization and generation of longer summaries (Kryściński et al., 2021; Huang et al., 2021; Zhang et al., 2022; Wu et al., 2021). Summaries in these settings differ considerably from the newswire summaries of past research efforts (Nallapati et al., 2016; Narayan et al., 2018): models now need to identify salient information and combine text from different parts of a significantly longer document while simultaneously ensuring that the generated text follows a coherent discourse structure.

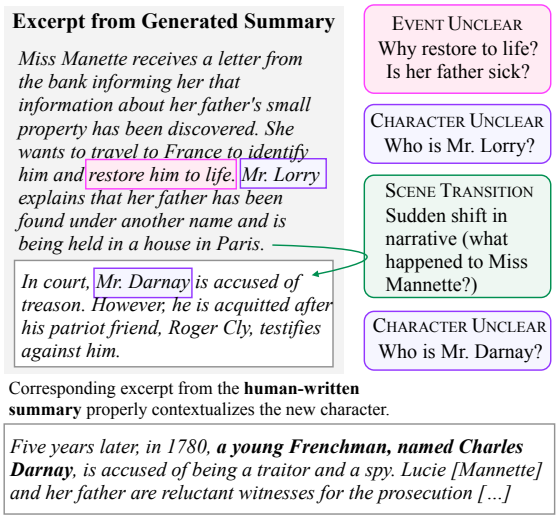

Figure 1: Excerpt from a generated book summary by OpenAI's 175B model (Wu et al., 2021). Individual segments do not follow a coherent structure and extra information is often needed to understand the narrative (e.g., *Who is Mr. Darnay?*). Compared to these, human-written summaries suitably contextualize new characters and events within the current narrative and setting.

This shift in the scope of the summarization task calls for a reexamination of the summarization evaluation framework. For short newswire summaries, Fabbri et al. (2021) showed that automated metrics are inadequate, and consequently, reporting results from a human annotation study has become the standard practice in the field. However, human evaluation is rarely done for longer summaries possibly due to the associated labour costs of reading and evaluating long text. It is also unclear whether A/B testing or Likert-scale based annotation frame-

[1]We release our SNAC dataset at https://coherence-annotation-summaries.herokuapp.com/

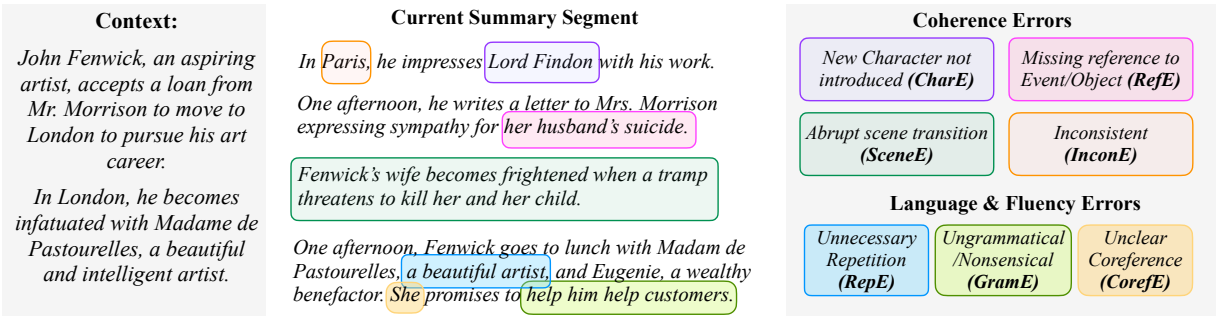

Figure 2: Error schema of our SNAC framework. Given context, i.e., the generated summary until that point, annotators identify error spans in the current summary segment. The figure shows error highlights according to our taxonomy for different options. We define two high-level error categories: (1) Coherence Errors that directly affect narrative understanding, and (2) Language Errors that measure other aspects, like grammar.

works transfer to long summary settings. Establishing human evaluation protocols is critical for reliably comparing different modeling approaches and measuring progress.

Recently, Wu et al. (2021) proposed a human-in-the-loop model for book summarization, reporting impressive performance. Their own analysis, however, revealed that although generated summaries contained *important* information from the books, they often read like a list of events stapled together without any coherent narrative structure (see Figure 1). We found similar artifacts in generated summaries from other recent narrative summarization models (Kryściński et al., 2021; Zhang et al., 2022). Now that models are so strong at generating fluent and on-topic sentences, the *coherence of the whole summary* becomes a first-order issue that must be evaluated in these new settings.

In this work, we introduce SNAC, a framework for collecting fine-grained annotations to evaluate **S**ummary **Na**rrative **C**oherence. We develop an error schema with 7 narrative error types grounded in actual errors made by current state-of-the-art summarization models (Wu et al., 2021; Zhang et al., 2022). Our fine-grained taxonomy allows annotators to explicitly state what kind of coherence error exists in a summary and pinpoint where it occurs. We show that such a fine-grained annotation framework is better suited for collecting crowd annotations compared to evaluating coherence holistically on a Likert scale.

We enlist crowd workers to collect a large-scale dataset of ~9.6k span-level error annotations in narrative summaries generated by current state-of-the-art summarization models (Wu et al., 2021; Zhang et al., 2022) on two different datasets: movie screenplays (Chen et al., 2022) and books (Kryściński et al., 2021). Our work is the first to characterize specific errors made by these systems and gaps that exist with human-written coherent summaries. While recent efforts have studied GPT-3 errors in open-ended generation (Dou et al., 2022), we show that these differ drastically from summarization errors and their taxonomies and findings are not transferable.

We also evaluate the performance of automatic coherence models, comparing synthetic data generation techniques (Moon et al., 2019; Shen et al., 2021) against SNAC annotations as training sources. Not only do models fine-tuned on SNAC outperform those trained on synthetic datasets, we find that they report higher recall than individual human annotators at identifying fine-grained coherence error categories.

We make our annotation tool publicly available; it can be easily customized to accommodate task-specific error schemas for annotation efforts along other dimensions of quality, e.g. fluency. We hope that our collected dataset and analysis will provide a foundation for further downstream applications such as improved longer summary evaluation, coherence-aware generation, post-correction of generated summaries, and others.

2 Long Narrative Summarization

Recent work has introduced a number of narrative summarization tasks, for books (Ladhak et al., 2020; Kryściński et al., 2021), screenplays (Chen et al., 2022; Papalampidi et al., 2020; Chen and Gimpel, 2021), and dialogue (Zhong et al., 2021). In our work, we study coherence errors in two domains, books and movie screenplays, although our taxonomy and annotation methodology are broadly applicable. The datasets and models evaluated are:

Domain	Dataset	Model	#sent	#word
News	XSum	BART/PEGASUS	1.0	19.2
News	CNN/DM	BART/PEGASUS	3.7	50.7
Book	BookSum	OpenAI 175B	33.9	572.7
Book	BookSum	OpenAI 6B	37.3	502.8
Movie	TRIPOD	SummˆN	40.5	765.5

Table 1: Comparison between generated summary lengths in narrative summarization and newswire.

Summary	R1	R2	RL	BS
OpenAI 175B	41.9	11.0	17.1	.51
+ Shuffled	41.9	11.0	15.6	.51
+ Repetition	44.7	10.6	17.2	.49
+ NE & bigram	42.8	10.1	16.3	.26
Human-written	45.8	12.5	17.9	.53

Table 2: ROUGE and BERTScore for BOOK-175B and several artificially corrupted versions. Results show that automatic metrics fail to penalize coherence errors.

1. **Books**: We evaluate summaries of books (Kryściński et al., 2021) generated by a GPT-3 based summarization model (Wu et al., 2021). We evaluate summaries from both the 175B and 6B versions of this model, denoted by BOOK-175B and BOOK-6B respectively.[2]

2. **Movie Screenplays**: We generate summaries for the movie scripts dataset, TRIPOD (Papalampidi et al., 2020), using the BART-based SummˆN model (Zhang et al., 2022).[3] We refer to these generated summaries as MOVIE-BART.

The average length statistics for these summaries are reported in Table 1. For comparison, statistics for the commonly used news domain summaries (Nallapati et al., 2016; Narayan et al., 2018) are also included; the majority of prior research in evaluation of evaluation metrics has focused on these datasets (Kryściński et al., 2019; Bhandari et al., 2020; Fabbri et al., 2021). The table clearly shows that there are substantial differences in the scale of these different domains.

We first explore whether existing approaches to evaluation can work well despite this difference, and establish that both existing metrics and human evaluation that have shown to work well for coherence in newswire do not work in our new settings.

2.1 Limitations of Automatic Metrics

Long document summarization research (Chen et al., 2022; Huang et al., 2021; Kryściński et al., 2021; Mao et al., 2021; Pang et al., 2022) has primarily relied on ROUGE scores to evaluate summaries. But do these capture narrative coherence?

We test this for long narrative summaries, using the BOOK-175B as a case study. Specifically, we test whether ROUGE or BERTScore (Zhang et al., 2019) can differentiate between actual generated summaries and their corrupted versions with artificially injected coherence errors. We introduce 3 types of coherence errors to generated summaries: (a) Random sentence **shuffling**, (b) **Repetition** of a randomly selected subset of sentences, and (c) Retaining only **named entities** in the summary and **top generated bigrams**; this set contains bigrams like *of the*, *that he*, etc. For an upper bound, we report these metrics for a different human-written summary for the same input book sampled from the BookSum dataset. More details are in Appendix B.

Automatic metrics fail to penalize coherence errors. Table 2 shows that both shuffling and repetition do not hurt ROUGE or BERTScore, despite introducing critical coherence errors in generated summaries. The *+NE & bigram* setting does lead to a significant drop in BERTScore as these summaries are no longer fully-formed sentences. However, even this trivial baseline reports ROUGE scores on par with the original BOOK-175B summaries, showing that ROUGE is easy to 'game' for this task. Finally, we see that human-written summaries, i.e., gold coherent summaries, only report +2 R2 and BERTScore improvement over artificially incoherent baselines. This clearly shows that these metrics are inadequate to measure coherence, or even overall quality, for long summaries.

2.2 Limitations of Human Annotation

Summary-level Likert-scale annotations are the most commonly used setup for collecting coherence in single-document news summarization research (Fabbri et al., 2021). Here, we run an analogous study for our longer narrative summaries.

[2]These generated summaries are publicly available at https://openaipublic.blob.core.windows.net/recursive-book-summ/website/index.html#booksum. We evaluate depth 1 summaries in our work.

[3]SummˆN is trained on TV episode screenplays. However, we noticed that TV episodes are not self-contained narratives and often refer to events or characters from previous episodes, making this an update summarization task which is harder to evaluate for coherence out of context. Therefore, we summarize movie scripts instead.

	News		Books
	Expert	Crowd	Crowd
Krippendorff's α	0.41**	0.48	0.19

Table 3: Summary-level agreement using the Likert scale. **Expert agreement after one round of annotations; this aligns with the crowd setting. News numbers taken from Fabbri et al. (2021).

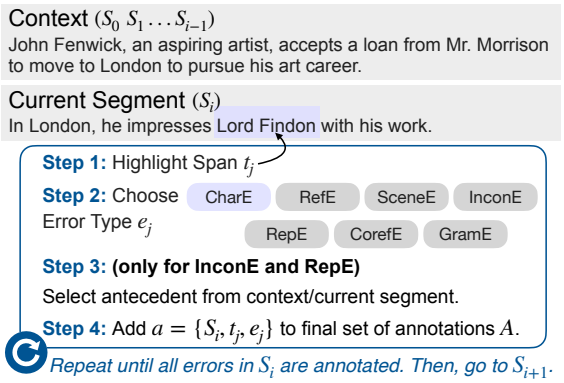

Figure 3: Workflow for annotating coherence errors in segment S_i with respect to the context, i.e. $S_0, S_1, ..., S_{i-1}$.

We ask 3 Mechanical Turk workers with prior experience in annotation for NLP tasks, specifically discourse analysis and text simplification, to rate the overall coherence of 100 generated summaries on a 5-point scale. Table 3 reports the observed agreement, measured by Kripendorff's α. We compare against Fabbri et al. (2021) who collect annotations for newswire summaries under a similar setup. We can see that annotations for longer narratives have much lower agreement compared to news summaries. We believe that this difference is due to two main reasons. First, newswire summaries are quite short and most models trained on these datasets generate highly extractive summaries (See et al., 2017; Goyal et al., 2022). This limits the scope of coherence errors within a single summary. For narratives on the other hand, it is difficult to devise annotation guidelines or reliably get a consensus on coherence for a 500+ word summary through a single value on a 5-point scale. Moreover, compared to news articles that generally refer to known events or people, book summaries contain fictional narratives, and therefore missing details here are harder to impute.

3 SNAC Annotation Methodology

Informed by our human study above, we design our annotation framework to achieve two main goals: 1) simplify the summary-level annotation task into smaller sub-tasks, and 2) provide a structured framework that allows annotators to specify the *type* of coherence error, instead of evaluating coherence holistically.

3.1 Task Workflow and Notation

We decompose the summary-level annotation task into smaller segment-level tasks: at each step, annotators evaluate a subpart of the summary, which is usually 2-4 sentences long. Let $S_0, S_1...S_N$ denote such text segments of a generated summary. While evaluating S_i, coherence judgments are made with respect to both the context $S_0, S_1...S_{i-1}$ and text within the segment S_i.

Our overall workflow for annotating errors in text segment S_i is shown in Figure 3. To annotate a single error, first, the annotators select the error span $t_j \in S_i$ and the coherence error type e_j (error taxonomy outlined in Section 3.2) to construct the error triple $a_j = (S_i, t_j, e_j)$. This process is repeated until all errors in segment S_i have been added, after which they proceed to the next text segment S_{i+1} for annotation. At the end of the annotation, workers produce the full set of annotations $A = \{a_j \; \forall j\}$ across all the text segments.

For OpenAI summaries, i.e. BOOK-175B and BOOK-6B, our segments come from boundaries present in the generated summaries. These text segments are an average of 2.7 sentences. For the MOVIE-BART summaries, we segment the summaries into chunks of 3 sentences.

3.2 Error Taxonomy

Coherence is defined in van Dijk (1977) as *"a semantic property of discourse, based on the interpretation of each individual sentence relative to the interpretation of other sentences."* Reinhart (1980) states three conditions for coherence: connectedness (cohesion), consistency, and relevance. Our error taxonomy is guided by these conditions while covering the broad range of coherence errors produced by state-of-the-art summarization models (Wu et al., 2021; Zhang et al., 2022).

We divide errors into two categories: a) **Coherence errors**: these measure whether the summary is well-structured and the events in the summary make narrative sense, and b) **Language errors**: these measure other aspects of the quality of generated text, such as grammar. While these do not come under the ambit of coherence errors, we found it useful to provide these additional error

types for crowd annotators to anchor other *badness* in text to.[4] We briefly outline error definitions below; see Figure 2 for illustrative examples.

3.2.1 Coherence Errors

New character without introduction (`CharE`) These refer to scenarios where a new person is introduced in the narrative without providing any background about the person, or their relation with other characters in the story. This violates condition 1 of coherence, i.e. connectedness. Note, however, that well-known people, e.g. Barack Obama, do not need to be explicitly introduced.[5]

Missing reference to an event or object (`RefE`) These refer to scenarios where an event or object is mentioned for the first time, but the phrasing strongly implies that it must have been introduced previously or that some context is missing to fully understand it. E.g., in Figure 2, the phrasing of *her husband's suicide* gives the strong impression that the reader is already aware of this event.

Abrupt scene transition (`SceneE`) These refer to errors where there is a sudden shift in the setting or narrative and are related to both connectedness and relevance. In less critical cases, these errors could be fixed by including a missing prepositional phrase, e.g. *Meanwhile in France, Fenwick's wife...* in Figure 2. Our generated summaries also include segments where the previous scene gets cut off when the wording strongly implied that more related events would follow; these **`SceneE`** errors cannot be easily fixed. We ask annotators to select whole sentences for this error type.

Inconsistency (`InconE`) These are directly aimed at errors that violate the second condition of coherence, i.e. contradicting other information in the *Context* or within the *Next Segment*. For these error spans, we additionally ask annotators to choose the previous span it is inconsistent with.

3.2.2 Language Errors

Repetition (`RepE`) These are used to detect content repetition. Similar to **`InconE`**, we additionally ask annotators to choose antecedent that contains the repeated information.

Ungrammatical or Nonsensical Text (`GramE`) These refer to text spans that have grammar errors. Also included in this category are cases where there are obvious model degenerations.

Unclear coreference (`CorefE`) These refer to errors where it is unclear who or what a pronoun is referring to. While sometimes requiring some clarity, we found that there errors rarely affected the overall narrative understanding unless they co-occured with **`GramE`**. Therefore, we do not include them in the coherence error category.

The version of definitions and task instructions given to the annotators is in Appendix D.

4 Data Collection

We collect annotations from two types of annotators: experts and crowd workers.

4.1 Expert Annotations

Expert annotations were collected from 3 authors who have previously published papers in text summarization and have experience engaging with model-generated text. Each annotator evaluated 10 book summaries, 5 each from BOOK-175B and BOOK-6B. This resulted in a dataset of ~700 span-level error annotations. Furthermore, we project span-level annotations to obtain binary coherent (no coherence error) and incoherent labels (at least one coherence error) at the sentence- and segment-levels. Table 4 provides statistics at these different levels of granularity.

We observed high inter-annotator agreement for expert annotators at both the sentence- and segment-levels (see Table 6). We used this dataset to train crowd workers in the next stage.

4.2 Crowd Annotations

Qualification We first launched a qualification task to recruit Mechanical Turk workers. The qualification was only made available to a subset of workers who had previously worked on other data annotation efforts for NLP tasks. In the qualification, we provided detailed instructions explaining the task workflow, interface, and error schema. Each annotator was asked to annotate 2 summaries from the books dataset; these summaries were chosen from the set of expert annotations. Workers were paid $12 for attempting this qualification.

We evaluated each worker's annotations against expert annotations and sent individual feedback.

[4] We want to evaluate generated summaries in isolation. Therefore, we do not test other summarization aspects that are dependant on the original text, e.g. salience or factuality.

[5] We special-cased this class of error because it was so frequent in our data. Our narratives are about fictional people in real-world settings, so places, organizations, and other named entity types are less likely to require explicit introduction.

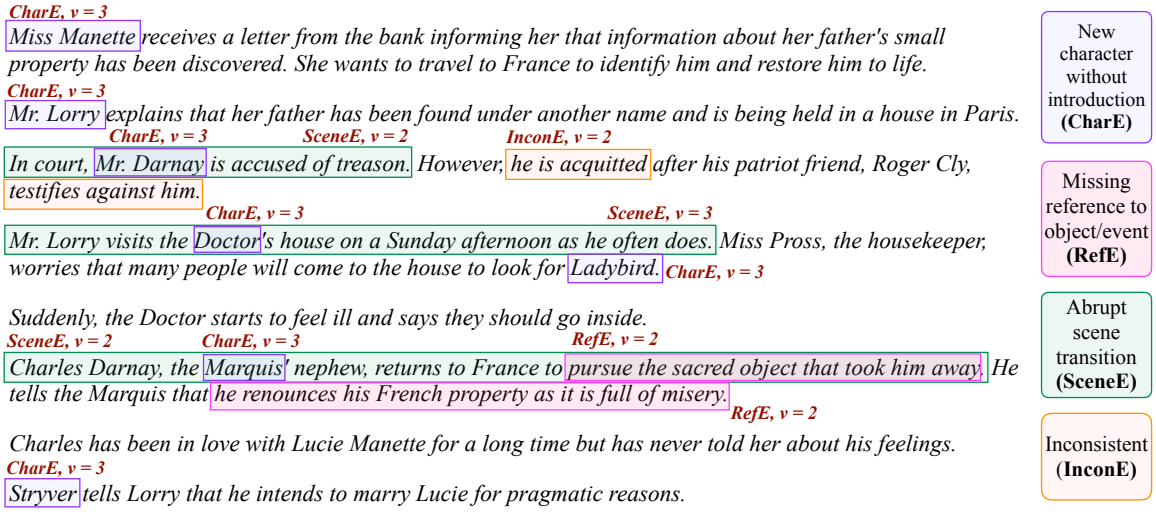

Figure 4: Example of span-level expert annotations for a BOOK-175B summary. The number of annotators who identified each span is denoted by v; for simplicity, we omit errors where $v = 1$. We found that annotators often identify overlapping coherence errors in the summary; this fine-grained understanding of the coherence issues in the model cannot be achieved by a summary-level coherence score.

Dataset	#summ	No. of Annotations Span	Sent	Seg
Expert Annotations				
BOOK-175B	5	323	173	111
BOOK-6B	5	401	174	66
Crowd Annotations				
BOOK-175B	55	3.1k	2.2k	1.1k
BOOK-6B	55	2.9k	2.2k	0.7k
MOVIE-BART	40	2.8k	1.8k	0.6k
Total	160	9.6k	6.6k	2.6k

Table 4: Statistics for expert and crowd annotations per level of granularity: span-, sentence- and segment-levels. Span-level annotations are multi-class, sentence- and segment-level have binary labels of coherence.

Among coherence errors, we observed that workers generally tended to disagree on **RefE**; each worker had a different calibration of which events or objects require more context to improve overall narrative understanding. Another common source of disagreement between workers and experts were **SceneE** errors. To help align their understanding with experts, we provided crowd workers with a complete set of expert annotations for a whole summary for reference.

Final Task We recruited 11 workers after the qualification task to annotate a total of 150 generated summaries. Each summary is annotated by 3 different annotators. Workers were paid an average of $12/hr for their work.

4.3 SNAC Dataset

Our resulting dataset consists of ~9.6k span-level annotations for coherence judgments, across 160 summaries. Dataset statistics for the entire collected dataset, including both expert and crowd annotations, are shown in Table 4.

A summary-wide expert annotation is SNAC is shown in Figure 4. Noticeably, **CharE** spans constitute the majority of errors; this observation is consistent throughout all datasets (see Figure 5). In fact, we saw that annotators tend to show higher recall and agreement over this error category. **SceneE** and **RefE** are the next two major error categories. Figure 4 annotations illustrate the two reasons for **SceneE** errors: (1) there is a sudden change in setting and characters, e.g. *Mr Lorry visits the...* and (2) the previous scene is abruptly cut off, e.g. *In court, Mr. Darnay ...*, where Ms. Mannette's story is unfinished.

We observed that worker annotations are high precision but low recall (**CharE** errors are an exception, workers have both high precision and recall for this category). This means that error spans identified by each worker tended to be an actual error, even when it was not detected by other annotators. Therefore, we combine annotations of all 3 annotators to construct the full SNAC dataset.

Open-Ended Generation $\neq$ Narrative Summarization In story completion, models are not required to cover all salient information from a document and only condition on past generated text;

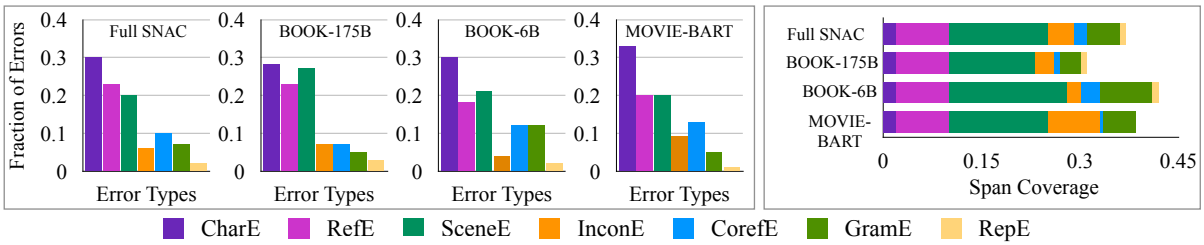

Figure 5: On the left, we show fraction of times a specific error type is detected for each individual dataset and their combination: **CharE**, **RefE** and **SceneE** errors constitute the majority of coherence errors. On the right, we show average fraction of error tokens belonging to each error-type in the SNAC dataset: smaller scale models (BOOK-6B and BART) have a much larger fraction of tokens identified as **GramE** errors compared to BOOK-175B.

generated open-ended summaries rarely diverge off-topic. Examples of GPT-3 generated stories in Figure 9 (Appendix A) show that these generate almost no **CharE**, **RefE** or **SceneE** errors that form the majority in SNAC, and instead mainly exhibit repetition. Therefore, research efforts that introduce fine-grained taxonomies for this task, e.g. SCARECROW (Dou et al., 2022), are not useful for summarization which needs to be independently studied.

Error Distributions Figure 5 shows the fraction of unique errors of each error type annotated across all datasets. As seen in Figure 4 annotations, the majority of the coherence errors are due to **CharE**, **RefE** or **SceneE**. The rightmost graph of Figure 5 shows the number of error tokens annotated (instead of numbers of errors) for each error type. We see that annotators mark a larger fraction of tokens in the BOOK-6B dataset as erroneous compared to BOOK-175B. The main difference comes from the difference in **SceneE** (annotators are instructed to select entire sentences) and **GramE**. As expected, for the smaller summarization models, i.e. GPT-3 6B and BART, a larger fraction of errors and error tokens are associated with language errors compared to GPT-3 175B. In fact, we noticed that workers were more likely to skip coherence error annotations, e.g. **RefE**, when these co-occur with **GramE** errors for these models, particularly on BOOK-6B.

Human annotators focus on language errors while assessing coherence holistically. We want to understand which aspects of the summary contribute to the summary-level coherence rating provided by crowd workers. We compute the correlation between the number of errors of each type with this overall coherence score assigned to summaries on a scale of 1-5 (described previously in Section 2.2).[6]

Error Type	Coherence	Language	Total
r	-0.26*	**-0.34***	-0.33*
Coherence Errors		Language Errors	
CharE	-0.22*	RepE	-0.21
RefE	**-0.29***	CorefE	-0.24*
SceneE	-0.05	GramE	**-0.25***
InconE	-0.09		

Table 5: Pearson Correlation between no. of errors and summary-level coherence score for error categories and their sub-types. Annotators tend to focus on grammar errors instead of coherence-specific errors while assigning overall summary-score. $*$: p-value < 0.05, according to a two-tailed test.

Table 5 outlines our results. First, it shows that the total number of errors is correlated with the overall coherence score, but annotators tend to weight language errors higher than coherence-specific errors. Surprisingly, we see negligible correlation between **SceneE** errors and overall score although these are a prominent distinguisher between generated summaries and human-written summaries. Amongst other error types, both **RefE** errors and **GramE** errors show relatively higher correlation. Although not directly evaluating coherence, Clark et al. (2021) report similar observations where annotators tend to focus on grammar errors while judging text quality.

4.4 Inter-Annotator Agreement

We first compute inter-annotator agreements at the **sentence- and segment-levels**. This allows us to perform an apples-to-apples comparison with

[6]We previously showed that annotators do not agree on overall summary ratings. However, this experiment differs in that each annotator's aggregated segment-level errors are correlated with *their own* summary-level judgment; here, agreement between annotators is not relevant, only document vs. segment-level consistency within a single annotator.

	Our Annotations				Newswire
	Expert		Crowd		Crowd
	Sent	Seg	Sent	Seg	Seg
Coherence	.77	.90	.59	.69	.49
Language	.33	.45	.22	.28	-

Table 6: Segment and sentence-level agreement, measured by Krippendorff's α for SNAC. Our dataset reports higher inter-annotator agreement compared to newswire summaries adapted to a similar setting.

Error	Krippendorff's α		Two-agree %	
	Expert	Crowd	Expert	Crowd
CharE	.91	.69	86	67
SceneE	.57	.30	62	35
RefE	.25 (.39)	.10 (22)	27 (39)	11 (23)
InconE	.18 (.29)	.13 (21)	20 (37)	14 (23)

Table 7: Token-level agreement for errors in the coherence sub-category. For **RefE** and **InconE**, we also report agreement (in brackets) after normalizing span boundaries for overlapping errors.

Fabbri et al. (2021) since the average length of newswire summaries is roughly equal to our segment length. We convert their 5-point Likert ratings into binary labels using the threshold that gives the best agreement score. We compare Krippendorff's α for SNAC and newswire in Table 6: we report high inter-annotator agreement at both the sentence- and the segment-level. Notably, the segment level agreement for our narrative texts is better than that of crowdworkers in the news domain.

Span-level analysis Next, we evaluate category-specific agreement between annotators at the span level. We report 2 metrics: 1) Krippendorff's α and 2) two-agree %; this is borrowed from Dou et al. (2022) and reports the percentage of tokens labeled as erroneous by at least one annotator that were also labelled by one or more additional annotators. For **RefE** and **InconE**, we noticed that small differences in span boundaries caused a significant drop in agreement, therefore, for these we also report metrics after normalizing span boundaries of overlapping spans to their union.

Table 7 outlines the agreement: for both expert and crowd annotators, we see high agreement for **CharE** and fair agreement for **SceneE** errors. On the other hand, lower agreement is observed for **RefE** category; this aligns with our observation that individual annotators may have low recall. Different annotators fundamentally have different notions of what extra information is critical for understanding the text.

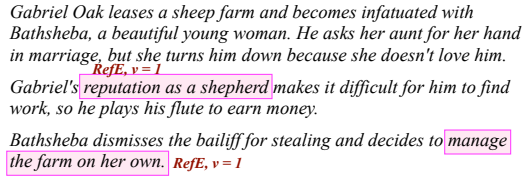

Figure 6: Examples of **RefE** errors identified by only one crowd annotator. Both these errors are instances where more information about the span can reasonably be expected to form a coherent narrative.

Similar overall results at the token-level are reported by Dou et al. (2022) for their error taxonomy: their error categories *Commonsense* and *Encyclopedic* report the lowest metrics, the two-agree % is as low as 20 and 12 respectively for 10 annotators. These numbers are expected to be much lower for 3 annotators, as in our setting.[7]

Figure 6 shows an example of a summary with low crowd agreement over the **RefE** errors (we omit all other identified errors in this figure). For the first highlight, it is reasonable to seek more clarity on why *Gabriel's reputation as a shepherd makes it difficult for him to find work* as it presupposes negative connotations associated with his profession that the reader is not privy to. The second highlight asserts that *Bathsheba* owns or works at 'the' farm as a known fact, which is information that has not been mentioned previously. Although both these are annotated by only one annotator, they qualify as **RefE** errors according to our definition.

5 Benchmarking Coherence Models

Human annotations can be prohibitively expensive to collect for long summaries. Can we train models to detect coherence errors in generated summaries?

Setup We formulate all models as sequence classifiers: given a context c and a sentence s, the goal is to classify whether s contains coherence errors. Similar to Section 4.3, we project span-level errors to a sentence-level gold coherence label $y^* \in \{0, 1\}$. Let $E = \{(e_j^*, t_j^*)\}$ denote the set of error types and corresponding spans in s.

We split the SNAC data into train (4.2k), dev

[7] We omit comparison with the Krippendorff's α reported in Dou et al. (2022) because they report *observed* agreement without normalizing by *expected* agreement. We re-compute their interannotator agreement on their dataset with normalization for a randomly selected subset of 3 annotators (comparable to our setting). This gives an average of 0.14 Krippendorff's α across all categories, with the bottom 5 categories reporting an average of 0.05 α.

(230) and test (1.8k) examples and evaluate all approaches on the test set.

Metrics First, we consider a **sentence-level binary classification** version of this task: can models correctly predict if a sentence contains coherence errors? In this case, our models take the form $P(y \mid c, s)$ where $y \in \{0, 1\}$.

We report precision, recall and F1 for all models. Note that the sentence-level y^{pred} judgment can be due to any of the 4 error types of their combination and does not tell us which of these error types are easier to detect. To answer this, we also report the error-wise recall under the binary setting: we assume $e_j^{\text{pred}} = 0$ if $y^{\text{pred}} = 0$ for all error types e_j. This overestimates the recall performance and can be viewed as an upper bound; a model that can only detect **`CharE`** may report non-zero recall for other errors if these co-occur with **`CharE`**. For fair comparison between different models, we report recall at the same precision level.

Second, we evaluate **fine-grained prediction**: can models identify the specific coherence error type and pinpoint the error span? In this case, our models predict $P(\mathbf{y} \mid c, s)$, where $\mathbf{y}$ is a bundle consisting of y and a set of error tuples $\{(e_j^{\text{pred}}, t_j^{\text{pred}})\}$ if $y = 0$.

Here, we report the precision, recall and F1 performance at correctly identifying the error type, i.e. $e_j^{\text{pred}} = e_j^* \ \forall e_j$. We also report $\circ\mathrm{v}$. computed as the fraction of times the predicted error span overlaps with the correct error span.

5.1 Models for Comparison

We compare the performances of three categories of models: (1) unsupervised (UNSUP) models. (2) Models trained on synthetic data targeting coherent errors (SYN): we follow prior work (Joty et al., 2018; Shen et al., 2021) and generate synthetic training data by introducing artificial coherence errors in reference text, specifically we use the BookSum dataset (Kryściński et al., 2021).[8] (3) Models fine-tuned on the SNAC data (FT). We describe these below:

(UNSUP) LM Perplexity We use GPT-2 (Radford et al., 2019) to obtain the probability of the sentence s, given the context c, by evaluating $P(s \mid c)$. The dev set is used to select a threshold τ_{LM} and obtain binary labels from these probabilities: predict an error if $P(s \mid c) < \tau_{LM}$.

[8] We ensure that there is zero overlap between our synthetic training data and the test set summaries used for evaluation.

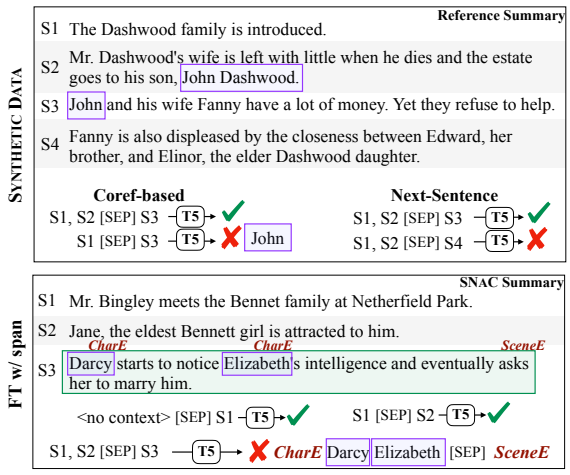

Figure 7: Methods to generate training data for the SYN and FT w/ span models. We fine-tune T5-Large for this binary classification task; for coref-based and FT w/ span, models also predict additional tokens.

(UNSUP) Entity Grid We construct entity grids (Barzilay and Lapata, 2005, 2008) for both predicted and gold summaries in order to compare their discourse structures. Using gold summaries in the BookSum dataset, we estimate the probabilities of syntactic role transitions between sentences, e.g. $p(S \to O)$, $p(S \to X)$, $p(O \to S)$, etc. Then, we score the coherence of a predicted summary s as the log probability of the transition from c^{-1}, i.e. the last sentence of context c, to sentence s: $w(c, s) = \sum_{e \in E} \log p(r(s, e) \mid r(c^{-1}, e))$. Here, E is the full set of entities in s and c^{-1} and $r(x, e)$ denotes the role of entity e in sentence x.

The SNAC dev set is used to select a threshold τ_{EG} and obtain binary labels from these scores: predict a coherence error if $w(c, s) < \tau_{EG}$.

(SYN) Coref-based This technique is designed to specifically target **`CharE`** and **`RefE`** errors. We run a coreference model (Lee et al., 2018) to extract coreferent chains in gold summaries. Let s_i, $s_{j>i}$ be sentences with the first and second mention of an entity. We derive non-coherent examples by setting $s = s_j$ and removing sentence s_i from the context, i.e. $c = s_1 s_2 ... s_{i-1} s_{i+1} ... s_{j-1}$ (see Figure 7). Conversely, for positive coherent training data, we retain the original context from the gold summaries, i.e. $c = s_1 s_2 ... s_i ... s_{j-1}$. We fine-tune T5-Large (Raffel et al., 2020) for binary classification $P(y \mid c, s)$ on these (y, c, s) triples; training data sizes and intrinsic performance are reported in Appendix C.

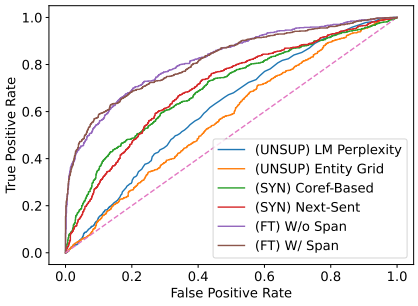

Figure 8: Performance of the baseline models and those trained on the SNAC test set. Models trained on human-annotated data outperform those trained on synthetically generated datasets.

(SYN) Next-Sentence This technique is designed to target **SceneE** errors and closely resembles the sentence-insertion method from prior work (Shen et al., 2021). Given context $c = s_1 s_2 ... s_i$, we obtain negative coherence examples by replacing the next sentence with another randomly sampled sentence from the remainder of the same summary, i.e. $s = s_j$, where $j > i + 1$. Figure 7 illustrates this. Positive examples are created by retaining the original summary completion, i.e. $s = s_{i+1}$. Again, we fine-tune T5-Large to model $P(y|c, s)$.

(FT) Model trained on SNAC data We consider two versions: (1) w/o span: trained to generate true/false reflecting the coherence of next sentence s, and 2) w/ span: trained to additionally predict the error category (e.g. *character* for **CharE**) and the corresponding error spans. Note that s can have errors belonging to multiple error categories, the model is trained to generate these in sequence. Figure 7 illustrates this. For **SceneE**, we omit span prediction as these are designed to incorporate the whole sentence. Similar to synthetic datasets, we fine-tune T5-Large for these tasks.

5.2 Results

Sentence-level binary classification Figure 8 shows the performance of the different models; the dotted line indicates random chance. First, we see that the entity-grid based approach performs poorly compared to all other neural approaches. Next, all trained models outperform the LM perplexity based model; language models aggregating token-level probabilities cannot detect coherence errors. Models trained on SNAC data outperform synthetic datasets which are the primary source of training data in prior work evaluating summary coherence. Our results clearly show that human annotated training data is necessary for training strong classifiers for automatic evaluation of coherence.

Model	CharE	SceneE	RefE	InconE	All
Coref-based	.61	.47	.48	.15	.43
Next-Sent	.31	.35	.32	.09	.27
FT w/o span	.89	.84	.64	.51	.73
FT w/ span	.90	.82	.58	.47	.70

Table 8: Sentence-level recall of different errors types for trained models at precision level $P = 0.7$. Models (except FT w/ span) do not predict the error category; here, we report the performance of the binary classification task irrespective of the predicted category.

Error	FT w/ span P	R	F1	ov.	Human P	R	F1	ov.
CharE	.79 (.86)	.81	.80	.98	.88	.71	.79	.98
SceneE	.35 (.58)	.49	.40	1.0	.58	.36	.44	1.0
RefE	.19 (.44)	.22	.21	.88	.31	.17	.22	.92
InconE	.25 (.25)	.02	.04	0.0	.29	.16	.20	.97

Table 9: Error-wise comparison between FT w/ span model and human annotators. Humans have higher precision while trained models report better recall across the top 3 error types.

Which error types are easier to detect for coherence models? We report category-wise recall for all models at the same precision level $P = 0.7$. Table 8 outlines our results. Both synthetic models report higher recall for the error category they were designed for. E.g., the coref-based method can detect **CharE** errors better than other error types. However, our FT models significantly outperform both synthetic approaches across all error types at thresholds with high precision performance. In particular, we observe high recall scores for **CharE** and **SceneE**.

Fine-grained prediction Only our FT w/ span model is trained to predict both the error category and the corresponding spans. Therefore, we compare its performance against humans annotators. For an apples-to-apples comparison, we reconstruct our test set by aggregating annotations of two randomly chosen annotators. This unfairly penalizes FT w/ span by introducing a mismatch between its train and test conditions, especially precision. Therefore, we also report precision scores on the original test set in brackets.[9]

Table 9 shows the fine-grained error detection capabilities of trained models and humans. As ob-

[9]Full set of results on the original test set derived from the complete SNAC annotations is included in Appendix C.

served during qualitative evaluation, humans are high-precision low-recall annotators. On the other hand, our FT w/ span model is trained on the aggregated annotations from three annotators and reports higher recall than humans. Consequently, the F1 scores for trained models are comparable to human performance except for the **InconE** category. We attribute this to the limited number of training examples of this category for the model to learn from.

Similar to previous analysis, we observe that models and humans report the best performance at detecting **CharE** errors. Interestingly, the trained model can identify both **SceneE** and **RefE** errors with higher recall compared to human annotators. Moreover, for the top three error types, trained models are successful at localizing error to specific spans, reporting high overlap scores.

6 Discussion

Our analysis of current narrative summarization models **reveals that these do not generate coherent narratives**; in fact, each generated summary contains ~40 coherence errors of varying degrees of severity. Moreover, both automatic and human approaches for coherence evaluation fail to reliably measure coherence. Our proposed framework SNAC addresses this gap and provides a protocol for training crowd workers and collecting large-scale coherence annotations.

However, we stop short of providing a prepackaged metric: which errors are more severe is application-dependent and overall error counts cannot be compared. Moreover, we observe that the severity of certain error categories is inherently subjective across people, particularly **RefE** errors. However, our error taxonomy and the resulting SNAC dataset allows us to conduct a detailed analysis of current coherence errors for different systems. This gives useful insights for designing improvements to summarization models along targeted error dimensions. We encourage future work to focus on fine-grained error annotations instead of sentence- or document-level annotations that do not provide similar actionable insights.

We recommend **fine-grained error modeling** for future coherence systems. While previous modeling has targeted document-level or sentence-level coherence, our models trained on SNAC data can detect span-level coherence errors, particularly **CharE** errors with high accuracy. This automatic error localization opens up future avenues of post-hoc error correction systems built on top of coherence models. Finally, although crowd annotators exhibited high precision, we saw that they often missed annotating coherence errors in text. Our high recall coherence models (compared to human annotators) can potentially be incorporated into the human evaluation pipeline to aid crowd workers during annotation.

7 Related Work

Coherence frameworks Inspired by Centering Theory (Grosz et al., 1995), Barzilay and Lapata (2005, 2008) proposed the entity-grid models to capture transitions of entity roles between sentences and measure coherence. This basic entity-grid model was further extended to incorporate non-head entities (Elsner and Charniak, 2011), discourse roles (Lin et al., 2011), and other improvements (Feng and Hirst, 2012; Feng et al., 2014) to better model text coherence. In recent years, neural variations of these (Guinaudeau and Strube, 2013; Nguyen and Joty, 2017; Joty et al., 2018) have been shown substantive improvements over the previous work. However, these models have been evaluated primarily on document-level coherence modeling on essay scoring tasks (Mesgar and Strube, 2018) or artificial sentence-ordering tasks (Shen et al., 2021). Their performance has not been evaluated on coherence errors produced by generation models, which differ substantially from these earlier settings.

Summarization Evaluation Automatic metrics such as BLEU, (Papineni et al., 2002), METEOR (Banerjee and Lavie, 2005), ROUGE, (Lin, 2004), BERTScore (Zhang et al., 2019), and others have been used to evaluate summarization and other generation models. However, Fabbri et al. (2021) showed that these automatic metrics show poor correlation with summary quality. Although human evaluation is widely considered the gold standard for generation tasks, recent work (Karpinska et al., 2021; Clark et al., 2021) demonstrated that humans are not reliable for evaluating strong models like GPT-3, across both A/B testing and Likert-scale based evaluation frameworks. To address this, Dou et al. (2022) proposed a fine-grained annotation framework and showed that task-specific error taxonomies and careful task design can avoid pitfalls of the previous human annotation studies. While these earlier studies have primarily focused

on open-ended generation errors, our work targets coherence errors made by narrative summarization models to address the gap in the current evaluation of such systems.

8 Conclusion

We introduce SNAC, a narrative coherence evaluation framework for summarization models. We develop an error taxonomy grounded in coherence errors made by current models and release span-level error annotations for 150 books and movie screenplay summaries. Our resulting data provides the first characterization of coherence errors in generated narrative summaries and allows us to train automatic classifiers to detect these errors. We make our annotation tool publicly available to support future research efforts in fine-grained error annotation for long text across other dimensions of generation quality.

Acknowledgments

Thanks to Eunsol Choi for providing feedback on this work, as well as our Mechanical Turk annotators for conducting the annotation. This work was partially supported by NSF Grant IIS-1814522, IIS-1850153, IIS-2107524, a gift from Amazon, and a gift from Salesforce Inc.

References

Satanjeev Banerjee and Alon Lavie. 2005. METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, pages 65–72.

Regina Barzilay and Mirella Lapata. 2005. Modeling local coherence: an entity-based approach. In *Proceedings of the 43rd Annual Meeting on Association for Computational Linguistics*, pages 141–148.

Regina Barzilay and Mirella Lapata. 2008. Modeling local coherence: An entity-based approach. *Computational Linguistics*, 34(1):1–34.

Manik Bhandari, Pranav Narayan Gour, Atabak Ashfaq, Pengfei Liu, and Graham Neubig. 2020. Re-evaluating evaluation in text summarization. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 9347–9359.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Mingda Chen, Zewei Chu, Sam Wiseman, and Kevin Gimpel. 2022. SummScreen: A dataset for abstractive screenplay summarization. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics*.

Mingda Chen and Kevin Gimpel. 2021. TVRecap: A Dataset for Generating Stories with Character Descriptions. *arXiv preprint arXiv:2109.08833*.

Elizabeth Clark, Tal August, Sofia Serrano, Nikita Haduong, Suchin Gururangan, and Noah A Smith. 2021. All that's 'human'is not gold: Evaluating human evaluation of generated text. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 7282–7296.

Yao Dou, Maxwell Forbes, Rik Koncel-Kedziorski, Noah A Smith, and Yejin Choi. 2022. Is GPT-3 Text Indistinguishable from Human Text? Scarecrow: A Framework for Scrutinizing Machine Text.

Micha Elsner and Eugene Charniak. 2011. Extending the entity grid with entity-specific features. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 125–129.

Alexander Richard Fabbri, Wojciech Kryściński, Bryan McCann, Caiming Xiong, Richard Socher, and Dragomir Radev. 2021. SummEval: Re-evaluating Summarization Evaluation. *Transactions of the Association for Computational Linguistics*, 9:391–409.

Vanessa Wei Feng and Graeme Hirst. 2012. Extending the entity-based coherence model with multiple ranks. In *Proceedings of the 13th Conference of the European Chapter of the Association for Computational Linguistics*, pages 315–324.

Vanessa Wei Feng, Ziheng Lin, and Graeme Hirst. 2014. The impact of deep hierarchical discourse structures in the evaluation of text coherence. In *Proceedings of COLING 2014, the 25th International Conference on Computational Linguistics: Technical Papers*, pages 940–949.

Tanya Goyal, Jiacheng Xu, Junyi Jessy Li, and Greg Durrett. 2022. Training Dynamics for Text Summarization Models. In *Findings of Association of Computational Linguistics*.

Barbara J Grosz, Scott Weinstein, and Aravind K Joshi. 1995. Centering: a framework for modeling the local coherence of discourse. *Computational Linguistics*, 21(2):203–225.

Camille Guinaudeau and Michael Strube. 2013. Graph-based local coherence modeling. In *Proceedings of the 51st Annual Meeting of the Association for*

Computational Linguistics (Volume 1: Long Papers), pages 93–103.

Luyang Huang, Shuyang Cao, Nikolaus Parulian, Heng Ji, and Lu Wang. 2021. Efficient Attentions for Long Document Summarization. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1419–1436.

Shafiq Joty, Muhammad Tasnim Mohiuddin, and Dat Tien Nguyen. 2018. Coherence modeling of asynchronous conversations: A neural entity grid approach. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 558–568.

Marzena Karpinska, Nader Akoury, and Mohit Iyyer. 2021. The perils of using mechanical turk to evaluate open-ended text generation. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 1265–1285.

Wojciech Kryściński, Nitish Shirish Keskar, Bryan McCann, Caiming Xiong, and Richard Socher. 2019. Neural text summarization: A critical evaluation. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 540–551.

Wojciech Kryściński, Nazneen Rajani, Divyansh Agarwal, Caiming Xiong, and Dragomir Radev. 2021. BookSum: A collection of datasets for long-form narrative summarization. *arXiv preprint arXiv:2105.08209*.

Faisal Ladhak, Bryan Li, Yaser Al-Onaizan, and Kathleen Mckeown. 2020. Exploring Content Selection in Summarization of Novel Chapters. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 5043–5054.

Kenton Lee, Luheng He, and Luke Zettlemoyer. 2018. Higher-order coreference resolution with coarse-to-fine inference. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 687–692.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880.

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

Ziheng Lin, Hwee Tou Ng, and Min-Yen Kan. 2011. Automatically evaluating text coherence using discourse relations. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies*, pages 997–1006.

Ziming Mao, Chen Henry Wu, Ansong Ni, Yusen Zhang, Rui Zhang, Tao Yu, Budhaditya Deb, Chenguang Zhu, Ahmed H Awadallah, and Dragomir Radev. 2021. DYLE: Dynamic Latent Extraction for Abstractive Long-Input Summarization. *arXiv preprint arXiv:2110.08168*.

Mohsen Mesgar and Michael Strube. 2018. A neural local coherence model for text quality assessment. In *Proceedings of the 2018 conference on empirical methods in natural language processing*, pages 4328–4339.

Han Cheol Moon, Muhammad Tasnim Mohiuddin, Shafiq Joty, and Chi Xu. 2019. A unified neural coherence model. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2262–2272.

Ramesh Nallapati, Bowen Zhou, Cicero dos Santos, Caglar Gulcehre, and Bing Xiang. 2016. Abstractive Text Summarization using Sequence-to-sequence RNNs and Beyond. In *Proceedings of The 20th SIGNLL Conference on Computational Natural Language Learning*, pages 280–290.

Shashi Narayan, Shay B Cohen, and Mirella Lapata. 2018. Don't Give Me the Details, Just the Summary! Topic-Aware Convolutional Neural Networks for Extreme Summarization. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 1797–1807.

Dat Tien Nguyen and Shafiq Joty. 2017. A neural local coherence model. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1320–1330.

Bo Pang, Erik Nijkamp, Wojciech Kryściński, Silvio Savarese, Yingbo Zhou, and Caiming Xiong. 2022. Long Document Summarization with Top-down and Bottom-up Inference. *arXiv preprint arXiv:2203.07586*.

Pinelopi Papalampidi, Frank Keller, Lea Frermann, and Mirella Lapata. 2020. Screenplay Summarization Using Latent Narrative Structure. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1920–1933.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 311–318.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever, et al. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

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

Tanya Reinhart. 1980. Conditions for text coherence. *Poetics today*, 1(4):161–180.

Abigail See, Peter J Liu, and Christopher D Manning. 2017. Get To The Point: Summarization with Pointer-Generator Networks. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1073–1083.

Aili Shen, Meladel Mistica, Bahar Salehi, Hang Li, Timothy Baldwin, and Jianzhong Qi. 2021. Evaluating document coherence modeling. *Transactions of the Association for Computational Linguistics*, 9:621–640.

Teun A. van Dijk. 1977. Text and context: explorations in the semantics and pragmatics of discourse. *Longman Linguistics Library, 21*.

Jeff Wu, Long Ouyang, Daniel M Ziegler, Nisan Stiennon, Ryan Lowe, Jan Leike, and Paul Christiano. 2021. Recursively summarizing books with human feedback. *arXiv preprint arXiv:2109.10862*.

Jingqing Zhang, Yao Zhao, Mohammad Saleh, and Peter Liu. 2020. PEGASUS: Pre-training with extracted gap-sentences for abstractive summarization. In *International Conference on Machine Learning*, pages 11328–11339. PMLR.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q Weinberger, and Yoav Artzi. 2019. BERTScore: Evaluating Text Generation with BERT. In *International Conference on Learning Representations*.

Yusen Zhang, Ansong Ni, Ziming Mao, Chen Henry Wu, Chenguang Zhu, Budhaditya Deb, Ahmed H Awadallah, Dragomir Radev, and Rui Zhang. 2022. Summˆ N: A Multi-Stage Summarization Framework for Long Input Dialogues and Documents. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics*.

Ming Zhong, Da Yin, Tao Yu, Ahmad Zaidi, Mutethia Mutuma, Rahul Jha, Ahmed Hassan, Asli Celikyilmaz, Yang Liu, Xipeng Qiu, et al. 2021. QMSum: A New Benchmark for Query-based Multi-domain Meeting Summarization. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5905–5921.

A Differences with Open-Ended Generation

Figure 9 shows examples of narrative completions obtained using the GPT-3 DaVinci model. We prompt GPT-3 with the first few sentences of three generated summaries and ask for a 500 work completion. The coherence errors contained in these model outputs are very different from the summarization setting. In particular, we see that here, the models hardly introduce any new characters (only *Mr. Greene* is introduced in the third example), and when it does, these are properly contextualized with the narrative. Furthermore, these models rarely generate **`RefE`** and generate no **`SceneE`** type of errors. In fact, repetition errors, shown in blue, dominate these narratives. Therefore, error taxonomies devised for these tasks, e.g. SCARECROW, are not useful for summarization settings.

B Limitations of Automatic Metrics

Here, we describe our artificial summary corruption methods from Section 2.1 in more details.

For **shuffling** errors, we derive a random permutation of all sentences in a BOOK-175B summary. This does not change the overall length of the generated summary.

For **repetition** errors, we randomly select 50% of the summary sentences and append a copy of the same sentence in place.

For ***+NE & bigram*** errors, we first extract the top 200 bigrams from the generated summaries in BOOK-175B. We noticed that these correspond to bigrams like *of the*, *that he*, *then he*, *in the*, etc. For each test set summary, we construct the corrupted summary by concatenating all named entities in the summary (appending each named entity as many times as it occurs in the original summary) and the top bigrams extracted from the testset-wide summaries.

Surprisingly, the ROUGE performance of the *+NE & bigram* baseline is similar to the original BOOK-175B summaries. This shows that ROUGE does not capture actual summary quality or coherence but instead only captures the presence or absence of named entities and commonly occurring bigrams.

C Benchmarking Coherence Models: Implementation details

Table 10 shows the training data sizes and the intrinsic performance of the synthetic dataset based

Prompt (Human-Written):
Complete this story in 500 words.
Miss Manette receives a letter from the bank informing her that information about her father's small property has been discovered.

Continuation by GPT-3 DaVinci:
The letter is from the bank. It reads:
"Dear Miss Manette, We regret to inform you that information about your father's small property has been discovered. The property was willed to you upon his death, but due to some legal complications, it has been tied up in the courts. We are currently working to resolve the matter, but in the meantime, we thought you should be aware of the situation. We will keep you updated on any developments. Sincerely, The Bank"

Miss Manette is surprised and disappointed to learn about the property. She had no idea her father even owned it. She wonders what legal complications are preventing her from accessing it.The bank assures her they are working to resolve the matter, and she decides to wait and see what happens. In the meantime, she is curious to learn more about the property and what her father's intentions were for her to have it. A few weeks later, the bank contacts Miss Manette again to let her know that the matter has been resolved and she is now the owner of the property. They give her the address and she is shocked to find that it is a small house in a rundown neighborhood. Despite the property's condition, Miss Manette is thrilled to finally have it and begins making plans to renovate it. She knows it will be a lot of work, but she is up for the challenge. With the help of her friends, she is able to transform the house into a cozy home that she can be proud of. Though it is not the property she originally envisioned, Miss Manette is happy to have a place to call her own.

Miss Manette was surprised and disappointed to learn about the property her father owned. She had no idea what legal complications were preventing her from accessing it, but the bank assured her they were working to resolve the matter. A few weeks later, the bank contacted Miss Manette again to let her know that the matter had been resolved and she was now the owner of the property. They gave her the address and she was shocked to find that it was a small house in a rundown neighborhood. Despite the property's condition, Miss Manette was thrilled to finally have it and began making plans to renovate it. She knew it would be a lot of work, but she was up for the challenge. With the help of her friends, she was able to transform the house into a cozy home that she could be proud of. Though it was not the property she originally envisioned, Miss Manette was happy to have a place to call her own.

Prompt (Human-Written):
Complete this story in 500 words.
The narrator, a Yale graduate, moves to New York to learn the bond business. He visits his second cousin, Daisy, and her husband, Tom, for dinner. During the dinner, Daisy mentions she wants to go back to Chicago the next day.

Continuation by GPT-3 DaVinci:
The narrator asks if he can go with her, and Daisy agrees. When they get back to Daisy and Tom's apartment, the narrator sees a man in the shadows who he later learns is Tom's mistress's husband. The next day, the narrator and Daisy take the train to Chicago. On the train, Daisy tells the narrator that she is pregnant and that Tom wants her to have an abortion. Daisy says she wants to keep the baby, but is afraid of what Tom will do if she does. The narrator offers to help Daisy, and they make a plan to meet in Chicago the following week.

The following week, the narrator meets Daisy in Chicago. They go to a hotel, and the narrator gets a room for Daisy. Daisy is hesitant to go into the room, but the narrator assures her that she will be safe. Daisy takes a bath, and the narrator reads to her from a book of poems. Later, Daisy falls asleep, and the narrator watches over her. In the morning, the narrator wakes Daisy and tells her it is time to go. They leave the hotel and take a taxi to the train station. Daisy thanks the narrator for his help, and they say goodbye. The narrator returns to New York and tells Tom what happened. Tom is angry and demands that the narrator leave. The narrator leaves, but not before telling Tom that Daisy is going to keep the baby. The narrator moves back to Chicago and meets Daisy again. Daisy is now married to a man named Gatsby, and they have a baby. The narrator is happy for Daisy and her new family.

The Yale graduate narrator moves to New York to learn the bond business. A few months after arriving, he visits his second cousin Daisy and her husband Tom for dinner. Daisy mentions wanting to go back to Chicago the next day, and the narrator asks if he can come along. The following day, the pair take the train to Chicago. On the ride, Daisy confides in the narrator that she is pregnant and Tom wants her to get an abortion. Daisy is torn because she wants to keep the baby, but is afraid of Tom's reaction. The narrator offers to help Daisy, and they make a plan to meet in Chicago the following week. The next week, the narrator meets Daisy in Chicago and takes her to a hotel. He gets her a room and Daisy takes a bath while the narrator reads to her from a book of poems. Daisy falls asleep and the narrator watches over her overnight. In the morning, the narrator wakes Daisy and tells her it is time to go. They leave the hotel and take a taxi to the train station. Daisy thanks the narrator for his help, and they say goodbye. The narrator returns to New York and tells Tom what happened. Tom is angry and demands that the narrator leave. The narrator leaves, but not before telling Tom that Daisy is going to keep the baby. The narrator moves back to Chicago and meets Daisy again. Daisy is now married to a man named Gatsby, and they have a baby. The narrator is happy for Daisy and her new family.

Prompt (Human-Written):
Complete this story in 500 words.
John Fenwick, an aspiring artist, accepts a loan from Mr. Morrison to move to London to pursue his art career. In London, he impresses several wealthy art collectors with his work.

Continuation by GPT-3 DaVinci:
One of them, Mr. Greene, purchases one of his paintings for a large sum of money. With the money from the sale, Fenwick is able to pay back Mr. Morrison and also has enough left over to buy materials for more paintings. He continues to sell his paintings and gains more recognition in the art world. John Fenwick was grateful when Mr. Morrison agreed to loan him the money to move to London so he could pursue his art career. He was determined to make a success of himself and worked hard to impress the wealthy art collectors in the city. His efforts paid off when one of his paintings was purchased by Mr. Greene for a large sum of money. Fenwick was able to pay back the loan and also had enough money left over to buy materials for more paintings. His talent and hard work continued to be rewarded with more sales and recognition in the art world.

He was on his way to becoming one of the most successful artists in London. One day, however, he was visited by Mr. Morrison, who had come to collect on the loan. Fenwick was surprised and disappointed to learn that the loan had never been forgiven and that he now owed a considerable amount of money, with interest. He was forced to sell more paintings to pay off the debt and was left with little money to live on. His dream of being a successful artist was over. Fenwick was heartbroken but he didn't give up. He continued to paint and eventually regained his place in the art world. He never forgot what Mr. Morrison did to him and he was determined to make sure that no one else would ever be treated so unfairly. Fenwick became a vocal advocate for artists' rights and worked to ensure that they were fairly compensated for their work. Thanks to his efforts, artists today are able to earn a living from their art and pursue their dreams. John Fenwick's story is one of talent, hard work, and determination. Despite being dealt a difficult hand, he was able to overcome adversity and make a lasting impact on the art world.

John Fenwick’s story is one of talent, hard work and determination. Despite being dealt a difficult hand, he was able to overcome adversity and make a lasting impact on the art world. When Fenwick was first starting out, he accepted a loan from Mr. Morrison to help him move to London so he could pursue his art career. He quickly made a name for himself in the art world and started impressing wealthy art collectors with his work. One of these collectors, Mr. Greene, ended up purchasing one of Fenwick’s paintings for a large sum of money.

Figure 9: Examples of open-ended story completion by the GPT-3 DaVinci model. The coherence errors observed under this setting (chiefly repetition errors, in blue) have little or no overlap with those from the summarization setting. Therefore, error taxonomies like SCARECROW that are devised for open-ended generated are not applicable to the summarization task.

coherence models (Section 5). We construct both our datasets with an equal number of positive and negative coherence examples. The results show that T5 learns to model the synthetic task with reasonable accuracy.

Method	#train	#dev	F1	Acc.
Coref-based	6.0k	920	.78	.77
Next-Sent	3.8k	880	.71	.74

Table 10: Dataset sizes and intrinsic performance of T5-Large models trained on synthetic datasets.

Table 11 shows the hyperparameters used for fine-tuning the T5-Large models on both synthetic training datasets and SNAC.

Computing Infrastructure	32GB NVIDIA V100 GPU
Max Input Seq Length	1024
Max Output Seq Length	80 (for FT w/ span)
Optimizer	Adam
Optimizer Params	$\beta = (0.9, 0.999), \epsilon = 10^{-8}$
Learning Rate Decay	Linear
Learning rate	1e-4
Batch size	8
Epochs	5

Table 11: Hyperparameters used for fine-tuning T5-Large on synthetic and SNAC train sets.

We compares human and model (FT w/ spans) performance in Table 9 on a modified test set created by combining annotations from 2 crowd workers. Here, in Table 12, we report results on the original test set that combines annotations from all 3 annotators.

D SNAC Error Schema

Here, we present the definitions of error types and illustrative examples provided to the crowd workers during training. These are also available at task website https://coherence-annotation-summaries.herokuapp.com/tutorial.

D.1 CharE

We call these **New Person not Introduced** in the task interface. We provide the illustrative example show in Figure 10 along with the following definition:

"*These refer to coherence errors where a new person is introduced into the narrative WITHOUT providing any background about the person, or their relation with other characters in the story. Note, however, that famous or well-known people do not need to be explicitly introduced.*"

Error	P	R	F1	ov.
CharE	.86	.74	.80	.99
SceneE	.58	.49	.53	1.0
RefE	.45	.25	.32	.87
InconE	.25	.01	.02	0.0

Table 12: Performance of the T5-Large model fine-tuned on the SNAC dataset at predicting the correct error type in each summary sentence. We also report the percentage of times the predicted span overlaps with the error span in the gold data.

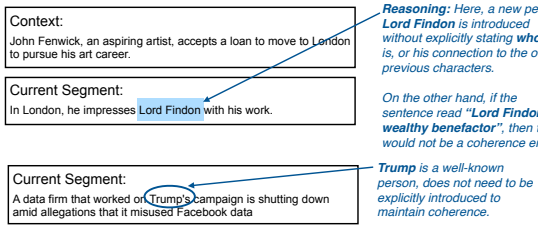

Figure 10: Illustration of **CharE** errors provided to crowd workers during training.

D.2 RefE

We call these **Missing Information about an Event/Object** in the task interface. We provide the illustrative example show in Figure 11 along with the following definition:

"*These refer to coherence errors where an event or object is mentioned for the first time, but the phrasing strongly implies some context is missing to understand this event/object and that it must have been introduced previously.*"

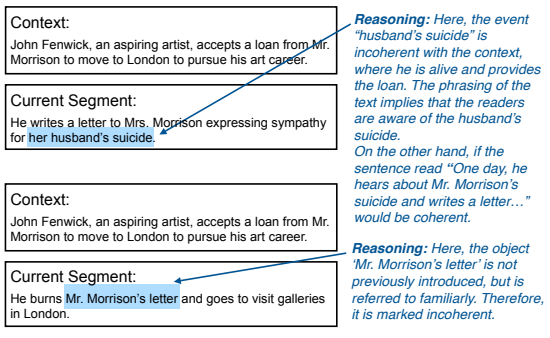

Figure 11: Illustration of **RefE** errors provided to crowd workers during training.

D.3 SceneE

These are called **Abrupt Transition from the Previous Scene** in the task interface. We provide the illustrative example show in Figure 12 along with the following definition:

"These refer to coherence errors where there is a sudden shift in the setting or the narrative in the story. These often happen in two scenarios:

1. *There is an abrupt change in the people/characters being discussed and/or an abrupt change in the surroundings/event.*

2. *Scenarios where the previous scene's phrasing strongly implies that more information/events are forthcoming, but the previous scene gets abruptly cut off and a completely new scene starts.*

Please choose full sentences as spans for this error type"

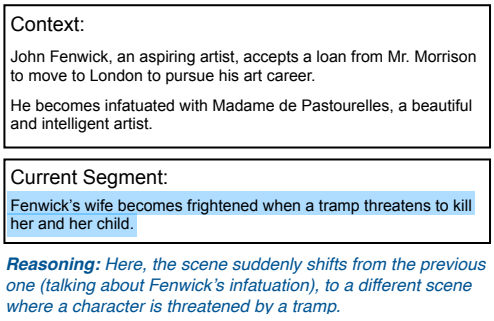

Figure 12: Illustration of **SceneE** errors provided to crowd workers during training.

D.4 InconE

Figure 13 shows an example of **Inconsistent** error shown to annotators.

"These refer to text spans that contradict previous content (either in the context or the next segment box itself.)

Note: You will also be asked to highlight the 'previous' span that is contradictory to the selected span. Highlighting this previous span (from either the context or the next segment box itself) will populate the relevant input box automatically."

D.5 CorefE

Figure 14 shows an example of **Unclear Coreference** provided to annotators.

"These refer to errors where it is unclear who/what a pronoun or refers to."

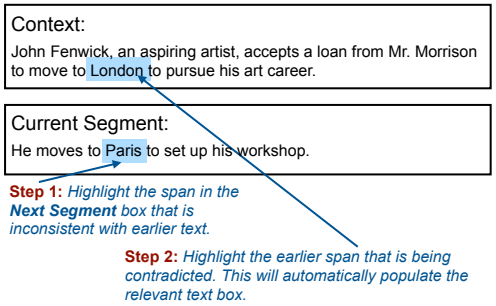

Figure 13: Illustration of **InconE** errors provided to crowd workers during training.

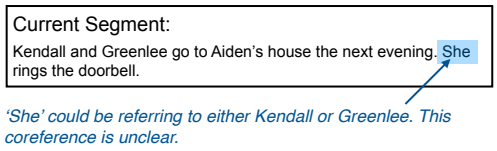

Figure 14: Illustration of **CorefE** errors provided to crowd workers during training.

D.6 RepE

Figure 15 shows an example of **Repetition** errors.

"These refer to spans where content is repeated.

Note: For these, you will also be asked to highlight the 'previous' span that contains the same text/content as the selected span. Highlighting this previous span (from either the context or the next segment box itself) will populate the relevant input box automatically."

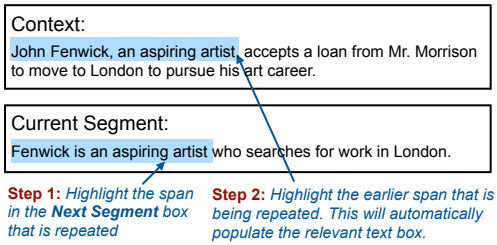

Figure 15: Illustration of **RepE** errors provided to crowd workers during training.

D.7 GramE

These are called **Ungrammatical/Nonsensical** in the interface.

"These refer to text spans that have grammar errors. Also included in this category are cases where there are obvious commonsense errors or the text does not make any sense at all."

D.8 Task Interface

Here, we show screenshots of our task interface. Figure 16 explains the basic task to the annotators. Figure 17 shows the detailed task workflow and the steps to annotate errors in a text segment. Figure 18 shows an example annotation with multiple coherence errors for reference.

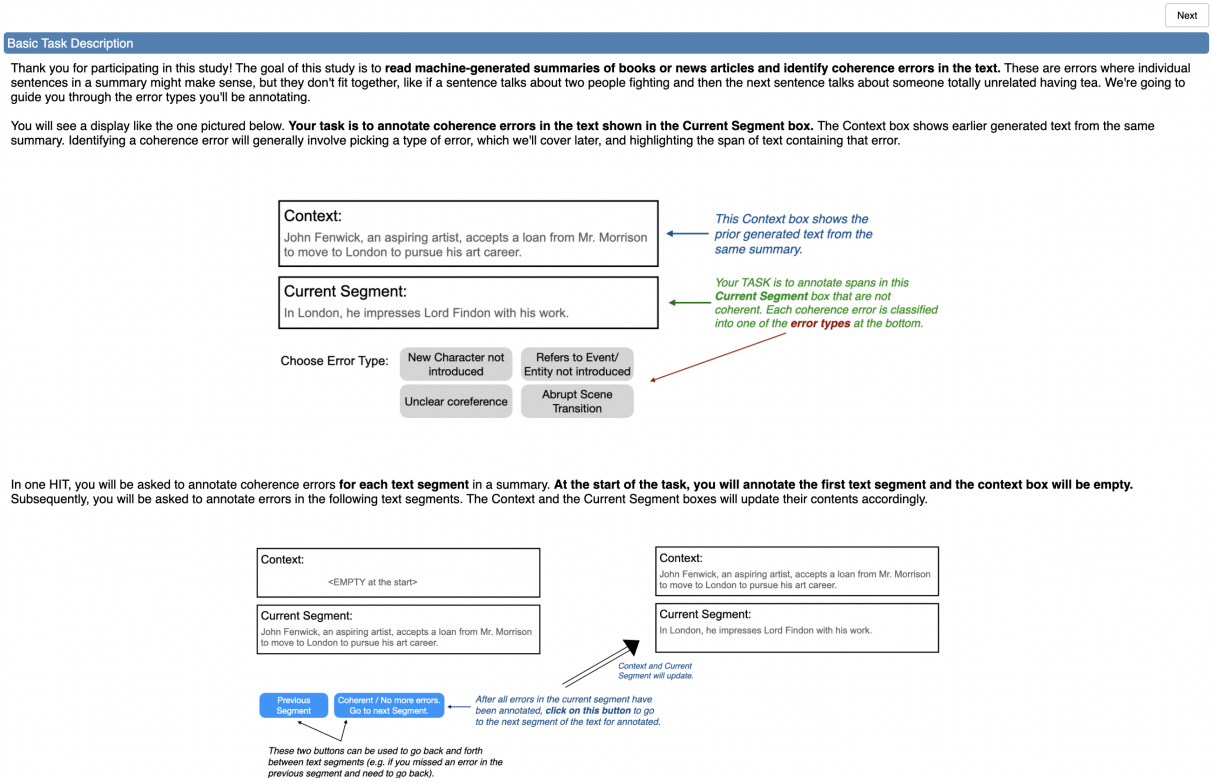

Figure 16: Screenshot of the first page of the tutorial provided to crowd annotators

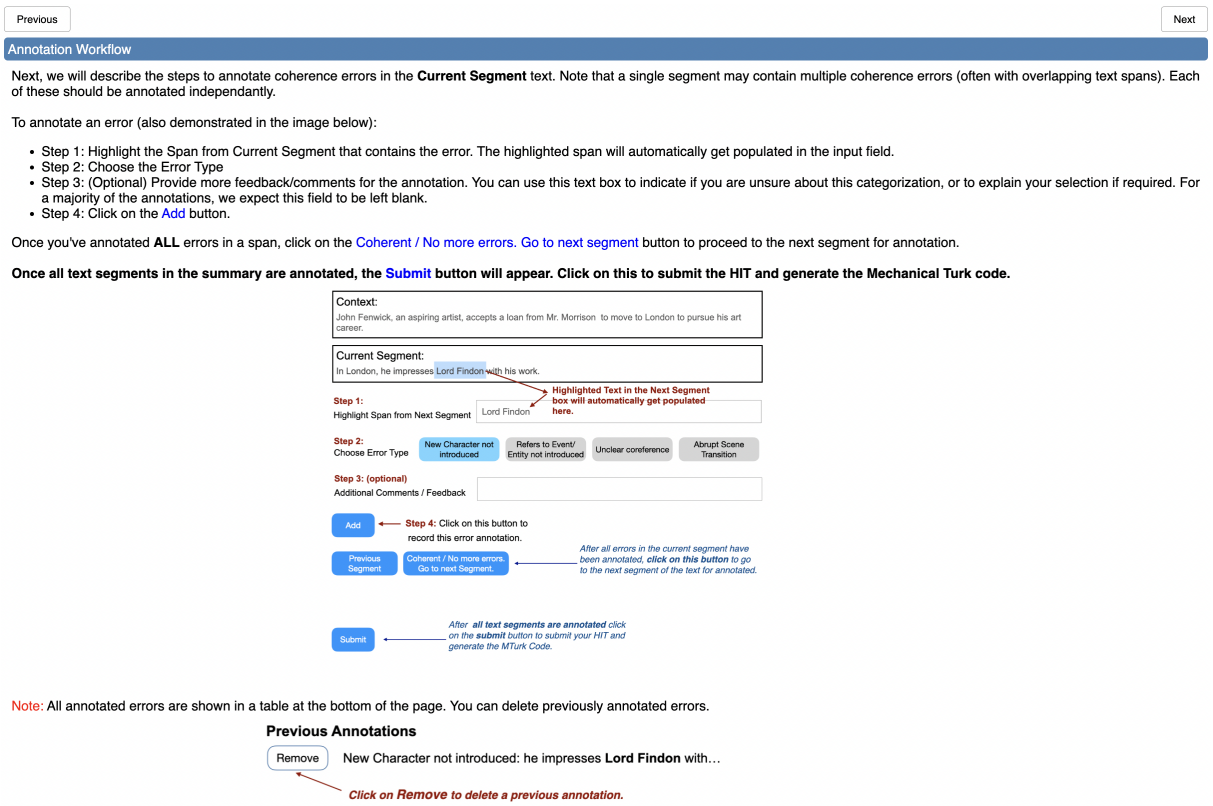

Figure 17: Screenshot of the second page of the tutorial provided to crowd annotators

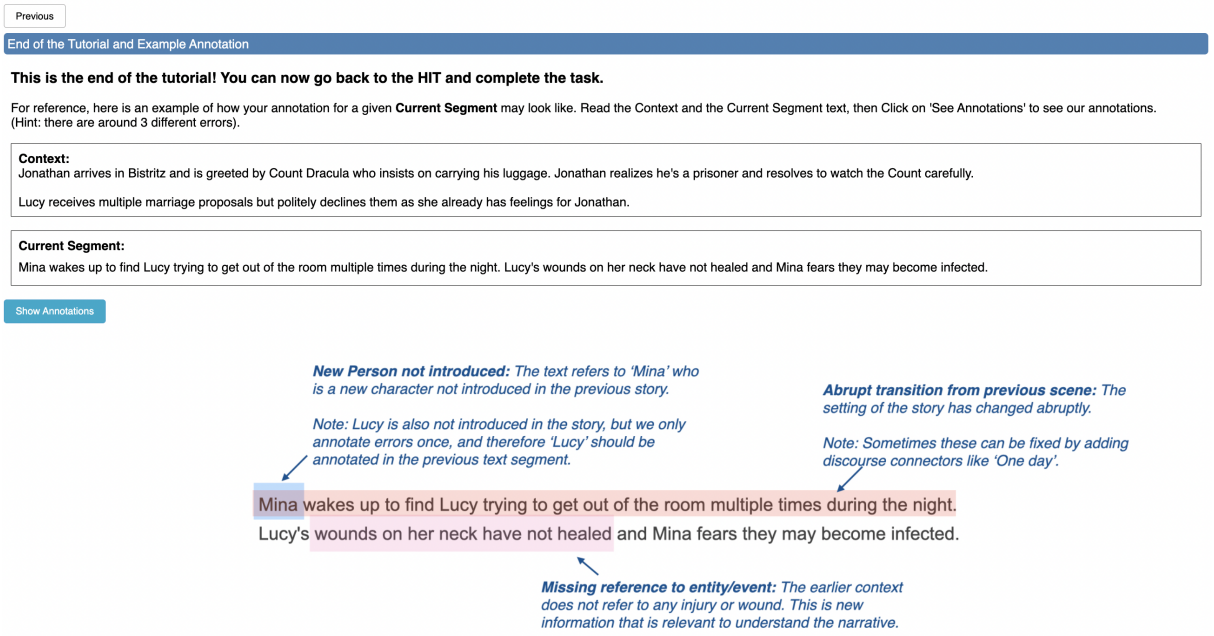

Figure 18: Screenshot of the last page of the tutorial provided to crowd annotators