

SNAC: Coherence Error Detection for Narrative Summarization

Tanya Goyal¹ Junyi Jessy Li² Greg Durrett¹

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. A long summary that appropriately covers the facets of that text must also present a coherent narrative, but current automatic and human evaluation methods fail to identify gaps in coherence. In this work, we introduce SNAC, a narrative coherence evaluation framework for fine-grained annotations of 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 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 crowdworkers. Furthermore, we show that the collected annotations allow us to benchmark past work in coherence modeling and train a strong classifier for automatically localizing coherence errors in generated summaries. 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, 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 extract salient information from different parts of a significantly

¹All collected annotations and models released at: <https://github.com/tagoyal/snac>.

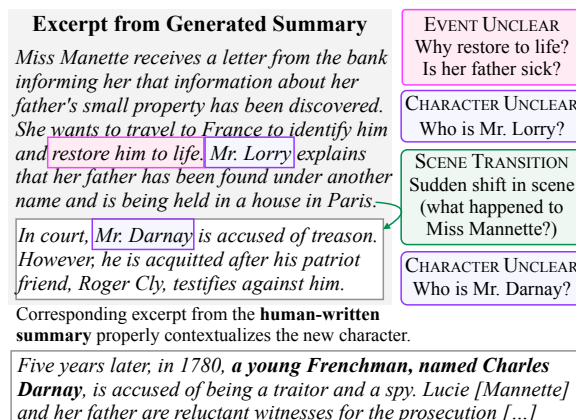


Figure 1: Excerpt from a generated book summary by OpenAI’s 175B model (Wu et al., 2021). Individual segments do not follow a coherent structure and extra information is often needed to understand the narrative.

longer document, and naïvely combining these in a much longer output is less likely to yield a summary with coherent discourse structure.

This shift in the scope of the summarization task calls for a reexamination of the summarization evaluation framework. Even for short newswire summaries, Fabbri et al. (2021) showed that automated metrics are inadequate, and consequently, reporting results from a human evaluation study has become the standard practice. However, human evaluation is rarely done for longer summaries possibly due to the associated labor costs of reading and evaluating long text. It is also unclear whether A/B testing or Likert-scale based annotation frameworks transfer to long summary settings. Establishing human evaluation protocols is critical for comparing different modeling approaches and measuring progress.

Recently, Wu et al. (2021) proposed a strong book summarization model but showed that although generated summaries covered *important* information from the books, they read like a list of events stapled together without any coherent narrative structure (see Figure 1). We found similar

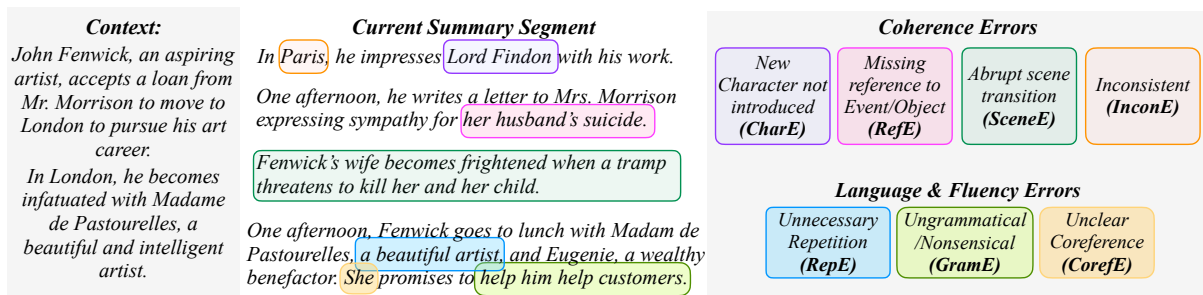


Figure 2: SNAC’s error schema. Given context, i.e., the generated summary until that point, annotators identify error spans in the current summary segment. We define two high-level error categories: (1) Coherence Errors that directly affect narrative understanding, and (2) Language Errors that measure other aspects, like grammar.

characteristics in other recent narrative summarization models (Kryściński et al., 2021; Zhang et al., 2022). Now that models are so good 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 **Summary Narrative Coherence**. We develop an error schema with 7 narrative error types grounded in actual errors made by current summarization models. 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 than a Likert scale-based holistic evaluation of coherence.

We enlist crowdworkers 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 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 errors in open-ended generation (Dou et al., 2022), these differ drastically from summarization errors and their taxonomies and findings are not transferable (see Appendix A).

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 also report higher recall

than individual human annotators at identifying fine-grained coherence error categories.

Our collected dataset and analysis provides a foundation for downstream applications such as better long summary evaluation, coherence-aware generation, and post-correction of summaries.

2 Long Narrative Summarization

We study coherence errors in two domains, books and movie screenplays, although our taxonomy and annotation methodology are broadly applicable.

Books We evaluate the depth 1 book summaries generated by a GPT-3 based model (Wu et al., 2021). We evaluate both its 175B and 6B versions, denoted by BOOK-175B and BOOK-6B respectively. On average, these are ~35 sentences long.

Movie Screenplays We generate summaries for the movie scripts dataset (Papalampidi et al., 2020) using the BART-based Summ^N model (Zhang et al., 2022).² These are ~40 sentences in length. We refer to them as MOVIE-BART.

The majority of prior research in evaluation of evaluation metrics (Kryściński et al., 2019; Bhandari et al., 2020; Fabbri et al., 2021) has focused on news summarization datasets (Nallapati et al., 2016; Narayan et al., 2018). However there exist substantial differences in the scale of news settings and narrative summaries: the former are considerably shorter at ~3 sentences per summary. We first explore whether existing approaches to evaluation can work well despite this difference.

²Summ^N is trained on TV episode screenplays. However, TV episodes are not self-contained narratives and often refer to events 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 (Expert)	News (Crowd)	Books (Crowd)
0.41**	0.48	0.19

Table 1: Summary-level agreement, measured by Krippendorff’s α . **Expert agreement after one round of annotations; this aligns with the crowd setting.

Limitations of Current Human Evaluation

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.

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 1 reports the observed agreement, measured by Krippendorff’s α . Compared to newswire summaries collected under a similar setup (Fabbri et al., 2021), annotations for longer narratives have a much lower agreement. This shows the difficulty in obtaining a consensus on coherence for a 500+ word summary through a single value on a 5-point scale.

In Appendix B, we further show that automatic metrics like ROUGE and BERTScore (Zhang et al., 2019) that are primarily used for evaluating long document summarization fail to penalize coherence errors in summaries. Better tools for both automatic and human evaluation are needed.

3 SNAC Annotation Methodology

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

To annotate a single error in S_i , 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 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. The outcome of this is shown in Figure 2.

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

3.2 Error Taxonomy

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 current models.

We divide errors into two categories: a) **Coherence errors**: these measure whether the summary is well-structured and 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 workers to anchor other “badness” in text to.

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 that well-known people, e.g. Barack Obama, do not need an introduction.³

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 occur where there is a sudden shift in the narrative and are

³We 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.

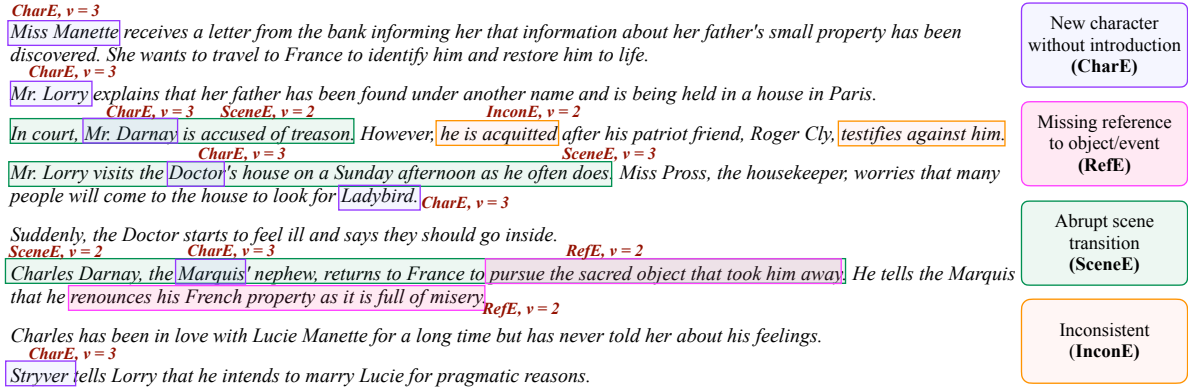


Figure 3: An example of expert annotations for a BOOK-175B summary (we only show coherence errors). The number of annotators who identified each span is denoted by v ; for simplicity, we omit $v = 1$ errors. We see that annotators often identify overlapping error spans; this fine-grained picture of coherence cannot be achieved by a summary-level score.

related to both connectedness and relevance. For these, we ask annotators to select whole sentences.

Inconsistency (InconE) These errors violate the second condition of coherence, i.e. contradicting other information in the *Context* or within the *Current Segment*. For these errors, we also ask annotators to choose the previous span it is inconsistent with.

3.2.2 Language Errors

Repetition (RepE) These are used to detect repetition. Similar to **InconE**, annotators also select the antecedent span with the repeated content.

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

Unclear coreference (CorefE) These refer to cases where it is unclear who or what a pronoun is referring to. While sometimes requiring extra clarity, we found that these errors rarely affected the overall narrative understanding unless they co-occurred 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 crowdworkers.

Type	Dataset	#summ	Span	Sent	Seg
Expert	BOOK-175B	5	323	173	111
	BOOK-6B	5	401	174	66
Crowd	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 2: 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.

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 2 provides statistics at these levels.

We observed high inter-annotator agreement for expert annotators at both the sentence- and segment-levels (see Table 4). We used this dataset to train crowdworkers in the next stage.

4.2 Crowd Annotations

We first launched a qualification task to recruit MTurk workers. The qualification was only made available to a subset of workers who had previously worked on other data annotation efforts for

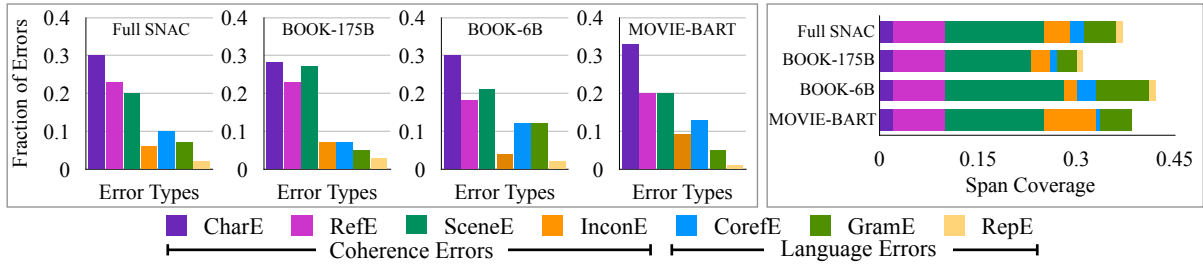


Figure 4: The left graph shows the fraction of times a specific error type is detected for each individual dataset: **CharE**, **RefE** and **SceneE** errors constitute the majority of coherence errors. The right graph shows the average fraction of tokens belonging to each error-type.

NLP tasks. It included detailed instructions explaining the task workflow, interface, and error schema. Each worker was asked to annotate 2 book summaries; 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 experts and sent individual feedback. 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 workers with a complete set of expert annotations for a whole summary for reference.

We recruited 11 workers after the qualification to annotate 150 generated summaries. Each summary was annotated by 3 different annotators. Workers were paid an average of \$12/hr.

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 2.

A summary-wide expert annotation is SNAC is shown in Figure 3. Noticeably, **CharE** spans constitute the majority of errors; this observation is consistent throughout all datasets. Annotators tend to show higher recall and agreement over this category. **SceneE** and **RefE** are the next two major error categories. The annotations also illustrate the two reasons for **SceneE**: 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 actual errors, even when they were not detected by other annotators. Therefore, we combine annotations of all 3 annotators to construct the full SNAC dataset.

Error Distributions Figure 4 shows the fraction of unique errors of each error type annotated across all datasets. As seen in Figure 3 annotations, the majority of the coherence errors are due to **CharE**, **RefE** or **SceneE**. The bottom graph of Figure 4 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 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** for these models, particularly on BOOK-6B.

Human annotators focus on language errors while assessing coherence holistically. To understand which aspects of a 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 the overall coherence score (Likert rating on a scale of 1-5, described previously in Section 2).⁴

⁴We previously showed that annotators do not agree on overall summary ratings. However, this experiment differs in

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

Table 3 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 with **SceneE** errors although these are a prominent distinguisher between generated 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.

Narrative Summarization \neq Open-Ended Generation

In story completion, models are not required to cover all salient information from a document and only condition on past generated text; generated open-ended summaries rarely diverge off-topic. Examples of GPT-3 generated stories in Figure 8 (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 directly applicable to summarization which needs to be independently studied.

4.4 Inter-Annotator Agreement

We first compute inter-annotator agreements at the **sentence- and segment-levels**. This allows for an apples-to-apples comparison with Fabbri et al. (2021) as the average length of news summaries is roughly equal to our segment length. We convert their 5-point Likert ratings into binary labels using

that each annotator’s aggregated segment-level errors are correlated with *their own* summary-level judgment; here, agreement between annotators is not relevant.

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

Table 4: 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 5: Token-level agreement for errors in the coherence sub-category. For **RefE** and **InconE**, we also report agreement (in parentheses) after normalizing span boundaries for overlapping errors.

the threshold that gives the best agreement score. We compare Krippendorff’s α for SNAC and news in Table 4: SNAC reports high inter-annotator agreement at both the sentence- and segment-level. Notably, this segment level agreement 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 two metrics: 1) Krippendorff’s α and 2) two-agree %; borrowed from Dou et al. (2022), this 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 5 outlines the agreement: for both expert and crowdworkers, we see high agreement for **CharE** and fair agreement for **SceneE**. On the other hand, lower agreement is observed for **RefE**; 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.

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

Gabriel Oak leases a sheep farm and becomes infatuated with Bathsheba, a beautiful young woman. He asks her aunt for her hand in marriage, but she turns him down because she doesn't love him. Gabriel's reputation as a shepherd makes it difficult for him to find work, so he plays his flute to earn money. Bathsheba dismisses the bailiff for stealing and decides to manage the farm on her own.

Figure 5: **RefE** errors identified by only one annotator.

% is as low as 20 and 12 respectively for 10 annotators. Note that we only have 3 annotators, so we expect our two-agree numbers to be much lower.⁵

Figure 5 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 only annotated by one annotator, these qualify as **RefE** errors according to our definition.

5 Detecting Coherence Errors

Setup We aim to see whether models can automatically detect our coherence errors. 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 SNAC into train (4.2k), dev (230) and test (1.8k) examples and evaluate 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 $P(y | c, s)$ where $y \in \{0, 1\}$. We report precision, recall and F1. Note that the sentence-level y^{pred} judgment can be due to any of the error types.

⁵We omit comparison with Krippendorff’s α reported in Dou et al. (2022) as 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 .

We next evaluate **fine-grained prediction**: can models identify the specific coherence error type and pinpoint the error span? In this case, our models predict $P(\mathbf{y} | c, s)$, where \mathbf{y} is a bundle consisting of y and a set of error tuples $\{(e_j^{\text{pred}}, t_j^{\text{pred}})\}$ if $y = 0$. 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 *ov.* computed as the fraction of times the predicted error span overlaps with the correct error span.

5.1 Models for Comparison

We compare performances of three types of models: (1) unsupervised (UNSUP). (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 on the BookSum dataset (Kryściński et al., 2021). We ensure zero overlap between this synthetic train set and the evaluation test set. (3) Models fine-tuned on the SNAC data (FT).

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

(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 of entities between sentences, e.g. $p(S \rightarrow O)$, $p(S \rightarrow X)$, $p(O \rightarrow S)$, 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 : $w(c, s) = \sum_{e \in E} \log p(r(s, e) | r(c^{-1}, e))$. 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

SYNTHETIC DATA		Reference Summary
S1	The Dashwood family is introduced.	
S2	Mr. Dashwood's wife is left with little when he dies and the estate goes to his son, John Dashwood.	coreferent
S3	John and his wife Fanny have a lot of money. Yet they refuse to help.	
S4	Fanny is also displeased by the closeness between Edward, her brother, and Elinor, the elder Dashwood daughter.	
		Coref-based
S1, S2 [SEP] S3	$\xrightarrow{\text{T5}}$ Yes	S1, S2 [SEP] S3 $\xrightarrow{\text{T5}}$ Yes
S1 [SEP] S3	$\xrightarrow{\text{T5}}$ No John	S1, S2 [SEP] S4 $\xrightarrow{\text{T5}}$ No
		Next-Sentence-based
S1, S2 [SEP] S3	$\xrightarrow{\text{T5}}$ Yes	S1, S2 [SEP] S3 $\xrightarrow{\text{T5}}$ Yes
S1 [SEP] S3	$\xrightarrow{\text{T5}}$ No John	S1, S2 [SEP] S4 $\xrightarrow{\text{T5}}$ No
FT w/ span		SNAC Summary
S1	Mr. Bingley meets the Bennet family at Netherfield Park.	
S2	Jane, the eldest Bennett girl is attracted to him.	
S3	Darcy starts to notice Elizabeth and asks her to marry him.	
	CharE	CharE
S1 [SEP] S2	$\xrightarrow{\text{T5}}$ Yes	
S1, S2 [SEP] S3	$\xrightarrow{\text{T5}}$ No CharE Darcy Elizabeth [sep] SceneE	

Figure 6: Training data generation, and T5 inputs and outputs for the SYN and FT w/ span models. The FT w/o span model only generates yes/no and not the specific category or span.

an entity. We derive non-coherent examples by setting $s = s_j$ and removing sentence s_i from the context, i.e. $c = s_1 s_2 \dots s_{i-1} s_{i+1} \dots s_{j-1}$ (shown in Figure 6). Conversely, for positive coherent training data, we retain the original context from the gold summaries, i.e. $c = s_1 s_2 \dots s_i \dots s_{j-1}$. We fine-tune T5-Large (Raffel et al., 2020) for binary classification $P(y | c, s)$ on these (y, c, s) triples; training data sizes and intrinsic performance are reported in Appendix C.

(SYN) Next-Sentence This method is designed to target **SceneE** errors and closely resembles the sentence insertion method from prior work (Shen et al., 2021). Given context $c = s_1 s_2 \dots s_i$, 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 $j > i + 1$. Positive examples are created by retaining the original summary completion, i.e. $s = s_{i+1}$. Figure 6 illustrates this. We fine-tune T5-Large to model $P(y | c, s)$.

(FT) Models trained on SNAC data We consider two versions: 1) w/o span: trained to generate yes/no reflecting the coherence of sentence s , and 2) w/ span: trained to additionally predict the error category (e.g. **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 6 illustrates this. For **SceneE**, we omit span prediction as these are designed to incorporate the whole sentence.

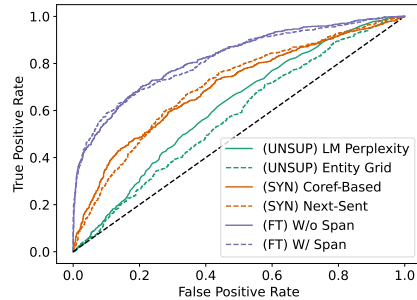


Figure 7: Performance of the different models on the SNAC test set. Models trained on SNAC outperform those trained on synthetically generated datasets.

Error	FT w/ span				Human			
	P	R	F1	ov.	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 6: Comparison between FT w/ span model and humans. Humans have higher precision while trained models report better recall across the top 3 error types.

Similar to SYN, we fine-tune T5-Large on these datasets.

5.2 Results

Sentence-level binary classification Figure 7 shows an ROC curve of different models; the dotted black line indicates random chance. It shows that the entity-grid approach performs poorly compared to all neural approaches. Next, all trained models outperform the LM perplexity model; language models aggregating token-level probabilities cannot detect coherence errors. Finally, models trained on SNAC outperform synthetic datasets which are the primary source of training data in prior coherence work. This shows that human annotations are needed to train strong coherence classifiers.

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 human annotators. For an apples-to-apples comparison, we re-construct 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. Full set of results on the original test set are in Appendix C.

Table 6 outlines the results. As observed during qualitative evaluation, the held-out human annotations are high precision and low recall. On the other hand, FT w/ span is trained on the aggregated annotations from three annotators and reports higher recall than humans. Consequently, its F1 scores are comparable to human performance except for **InconE**. We attribute this to the limited number of training examples of this category.

Similar to previous analysis, we observe that models and humans report the best performance at detecting **CharE**. Interestingly, the trained model can identify both **SceneE** and **RefE** with higher recall compared to human annotators. For these 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 ~30 coherence errors of varying degrees of severity. Moreover, both automatic and human approaches for coherence evaluation fail to reliably measure coherence. SNAC addresses this gap.

However, we stop short of providing a prepackaged metric: which errors are more severe is application-dependent and subjective, and overall error counts cannot be compared. We encourage future work to focus on fine-grained error annotations, like those we present here, instead of sentence- or document-level annotations that do not provide actionable insights. We also recommend fine-grained error modeling for future coherence systems as well. While previous modeling has targeted document- 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.

7 Related Work

Coherence frameworks Inspired by Centering Theory (Grosz et al., 1995), Barzilay and Lapata (2005, 2008) proposed entity-grid models to measure coherence through transitions of entity roles. This 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), including neural variations (Guinaudeau and Strube, 2013; Nguyen and Joty, 2017; Joty et al., 2018) to better model text coherence. However, these models have been evaluated primarily on document-level essay scoring tasks (Mesgar and Strube, 2018) or artificial sentence-ordering tasks (Shen et al., 2021), and not on model-generated coherence errors.

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, but Fabbri et al. (2021) showed that these correlate poorly with summary quality. Human evaluation is widely considered the gold standard for generation tasks, however, recent work (Karpinska et al., 2021; Clark et al., 2021) demonstrated that humans are not reliable for evaluating strong models like GPT-3.

8 Conclusion

We introduce SNAC, a narrative coherence evaluation framework for long summaries. We develop an error taxonomy grounded in coherence errors made by current models and annotate data to provide the first characterization of such errors in narrative summaries. We also make our annotation tool publicly available to support future research efforts.

9 Limitations

Although we view this work as an important step towards better understanding and evaluation of coherence in summaries, we acknowledge there is much more to do here. In this work, we only collect annotations and analyze coherence errors in summaries of English language books and movie screenplays. Our proposed taxonomy may not cover errors made by text summarization models for other languages and our trained models and analysis are English-specific.

Moreover, some of these books summarized were written decades ago and may reflect the societal biases of those times, which could conceivably bias our trained error detection models. In this work, we use the text from the model generated summaries as is and do not perform any filtering.

Finally, our work studies generated summaries for long narrative text. While we believe that our taxonomy is generalizable to other types of narrative text, we do not investigate whether it covers

other domains involving summarization of long documents, such as government report summarization (Huang et al., 2021) or meeting summarization (Zhong et al., 2021).

Acknowledgments

Thanks to Eunsol Choi and Michael Strube for providing feedback on this work, as well as our Mechanical Turk annotators for conducting the annotation. This project was partially supported by Good Systems,⁶ a UT Austin Grand Challenge to develop responsible AI technologies, a grant from the UT Austin Office of the Vice President for Research through the “Creating Connections for National Security Research Grants” program, a grant from Open Philanthropy, NSF grants IIS-2107524, IIS-2145479, and gifts from Salesforce, Amazon, and Adobe.

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*.
- 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. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics*.
- 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.
- 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 Attention 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.

⁶<https://goodsystems.utexas.edu/>

- 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*.
- 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.
- 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.
- 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. Summ^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 Narrative Summarization \neq Open-Ended Generation

In Section 4.3, we noted that narrative summarization exhibits substantially different errors than open-ended text generation tasks like story generation or story completion, hence the need for our new taxonomy. We show examples of generated stories using the GPT-3 DaVinci in Figure 8. We prompt the GPT-3 text-davinci-002 model with the first few sentences of three generated summaries and ask for a 500-word completion. The coherence errors contained in these model outputs are very different from those in our narrative summarization setting. In particular, the stories hardly introduce any new characters (only *Mr. Greene* is introduced in the third example), and when they do, 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 (Dou et al., 2022), are not useful for summarization settings which needs to be independently studied.

B 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 dataset 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:

1. Random **shuffling** using a random permutation of all sentences in a BOOK-175B summary. This does not change the overall length of the generated summary.
2. **Repetition** of a randomly selected subset of sentences. We randomly sample 50% of the sentences to repeat, all other sentences only occur once.
3. Retaining only **named entities** in the summary and **top generated bigrams**. We first extract the top 200 bigrams from the generated summaries in BOOK-175B, which include frequent bigrams like *of the, that he, then he, in the,* etc. For each test set summary, we construct a 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.

For an upper bound, we also report metrics for a different human-written summary for the same input book sampled from the BookSum dataset.

Automatic metrics fail to penalize coherence errors. Table 7 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 points of improvement in R2 and BERTScore over artificially incoherent baselines. This clearly shows that these metrics are inadequate to mea-

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 8: Examples of open-ended story completion by the GPT-3 text-davinci-002 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.

Summary	R1	R2	RL	BERTScore
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 7: ROUGE and BERTScore for BOOK-175B and several artificially corrupted versions. Results show that automatic metrics fail to penalize coherence errors.

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

Table 8: Dataset sizes and intrinsic performance of T5-Large models trained on synthetic datasets.

sure coherence, or even overall quality, for long summaries.

C Detecting Coherence Errors: Details and Additional Results

C.1 Models trained on synthetic data (SYN)

Table 8 shows the training data and development data size, as well as the intrinsic performance of these synthetic dataset-based coherence models on this development set. 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. We do not expect the models to perform perfectly, as the synthetic data may have false positives (examples constructed to exhibit errors that are actually coherent).

C.2 Implementation Details

Table 9 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 9: Hyperparameters used for fine-tuning T5-Large on synthetic and SNAC train sets.

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 10: Sentence-level recall of different errors types. Models (except FT w/ span) do not predict the error category; here, we treat these methods as binary classifiers and compute recalls as described in Appendix C.3.

C.3 Additional Results

Sentence-level binary classification In Section 5, we reported sentence-level binary classification results for all models. However, the sentence-level y^{pred} judgment in that setting can be due to any of the 4 error types or their combination and binary classification metrics do not tell us which of these error types are easier to detect.

To answer this, we compute 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 ; that is, a prediction of a binary error counts as detecting an error of any type in that sentence. 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 category-wise recall for all models at the same precision level $P = 0.7$. Table 10 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 In Table 6, we compared human and model (FT w/ spans) performance on a modified test set created by combining annotations from 2 crowdworkers. This unfairly penalized the trained models, which may have slightly higher recall due to being trained on annotations from 3 crowdworkers. In Table 11, we report results on the original test set that combines annotations from all 3 annotators.

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 11: 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.

D SNAC Error Types and Task Interface

The definitions of error types and illustrative examples provided to the crowdworkers during training are outlined here.

D.1 CharE

We call these **New Person not Introduced** in the task interface. We provide the illustrative example show in Figure 9 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.”

Figure 9: Illustration of **CharE** errors provided to crowdworkers 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 10 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 10: Illustration of **RefE** errors provided to crowdworkers 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 11 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 11: Illustration of **SceneE** errors provided to crowdworkers during training.

D.4 InconE

Figure 12 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.”

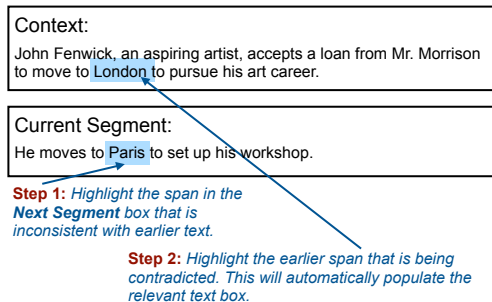


Figure 12: Illustration of **InconE** errors provided to crowdworkers during training.

D.5 CorefE

Figure 13 shows an example of **Unclear Coreference** provided to annotators.

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

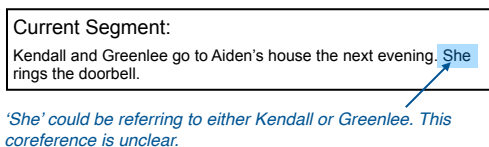


Figure 13: Illustration of **CorefE** errors provided to crowdworkers during training.

D.6 RepE

Figure 14 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.”

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

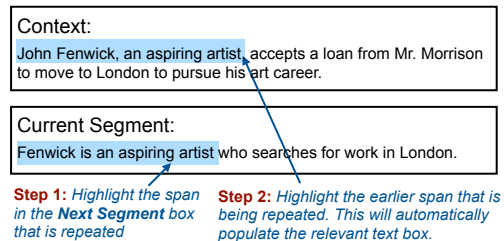


Figure 14: Illustration of **RepE** errors provided to crowdworkers during training.

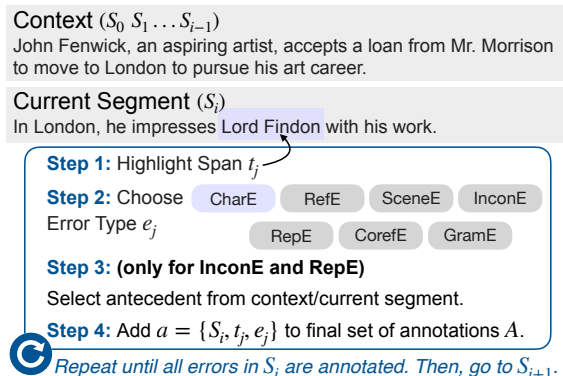


Figure 15: Workflow for annotating coherence errors in segment S_i with respect to the context, i.e. S_0, S_1, \dots, S_{i-1} .

where there are obvious commonsense errors or the text does not make any sense at all.”

D.8 Task Interface

In Section 3.1, we described the annotation work for the SNAC framework. Figure 15 visually illustrates this overall workflow for annotating errors in segment S_i . A screenshot of the actual task interface is shown in Figure 16.

We also include screenshots of our task instructions. Figure 17 explains the basic task to the annotators. Figure 18 shows the detailed task workflow and the steps to annotate errors in a text segment. Figure 19 shows an example annotation with multiple coherence errors for reference.

E Datasheet for SNAC

E.1 Motivation for Dataset Creation

Why was the dataset created? Despite recent interest in long document summarization research and generation of long narrative summaries (Kryściński et al., 2021; Zhang et al., 2022; Mao et al., 2021; Wu et al., 2021), we lack evaluation frameworks to compare these approaches and measure progress. Current automatic and human evaluation

methods fail to identify gaps in narrative coherence and are not suited for evaluating long summaries. Our SNAC dataset and annotation framework releases a large-scale dataset of fine-grained coherence annotations and establishes a protocol for eliciting such annotations from crowdworkers. This provides a foundation for future research efforts in this area.

Has the dataset been used already? At the time of submission, the dataset has only been used in the current paper for analysis of generation errors made by current state-of-the-art summarization models and for training automatic coherence detection models.

Who funded the dataset? We withhold this information to maintain anonymity but will include it upon publication.

E.2 Dataset Composition

What are the instances? Each instance in this dataset is a model generated summary from either the book or the movie domain. All summaries are in the English language.

How many instances are there? Our dataset contains annotations for 160 generated summaries (including both expert and crowd annotations).

What data does each instance consist of? Each instance contains multiple span-level highlights corresponding to coherence errors, each of which is tagged with a specific error category.

Does the data rely on external sources? Yes. For the book datasets, we annotate summaries from the publicly available model outputs released by Wu et al. (2021). For movies, we generate summaries using the Summ^N model (Zhang et al., 2022) on the publicly available TRIPOD dataset (Papalampidi et al., 2020).

Are there recommended data splits or evaluation measures? We will include the recommended training, development, and test splits for our annotations with the dataset release. The statistics for the data splits are outlined in Section 5.

E.2.1 Data Collection Process

Who was involved in the collection process and what were their roles? For expert annotations, 3 authors of the paper with experience in engaging with model-generated text annotated 10 book summaries. To recruit crowd annotators, we launched

a qualification task on Mechanical Turk. After this qualification, 11 workers were asked to annotate 150 summaries.

How was the dataset collected? Given a generated summary, annotators were asked to select span highlights that correspond with coherence errors and categorize the type of that error. We provided all annotators with detailed instructions describing the task interface, error type definitions as well as the overall workflow.

Over what time frame was the data collected? The dataset was collected over the months of March and April 2022.

Does the dataset contain all possible instances? No, we only annotate narrative summaries from two summarization models on two domains (movies and books). Moreover, our dataset only contains English language summaries.

If the dataset is a sample, then what is the population? The dataset is a subset of generated summaries produced by state-of-the-art summarization models on narratives like books or movie screenplays.

E.3 Data Preprocessing

What preprocessing/cleaning was done? We fix sentence and word boundaries for highlighted spans from crowd annotations.

Was the raw data saved in addition to the cleaned data? Yes.

Does this dataset collection/preprocessing procedure achieve the initial motivation? Yes. This dataset serves as a large-scale collection of annotated coherence errors and provides the first characterization of such errors in long narrative summaries.

E.4 Dataset Distribution

How is the dataset distributed? Our dataset is publicly released at this link: <https://github.com/tagoyal/snac>.

When was it released? The dataset was released in October, 2022.

What license (if any) is it distributed under? The dataset is released under the CC BY-SA 4.0 license.⁷

⁷<https://creativecommons.org/licenses/by-sa/4.0/legalcode>

Who is supporting and maintaining the dataset?

This dataset is maintained by authors of this paper.

E.5 Legal and Ethical Considerations

Were workers told what the dataset would be used for and did they consent? Crowdworkers were aware that their responses were being collected as part of a research study on analyzing coherence errors in narrative text. The Amazon Mechanical Turk Participation Agreement permits the use of their annotated responses for this work. We do not release any personal information, e.g. worker IDs, of the crowdworkers.

If it relates to people, could this dataset expose people to harm or legal action? No.

If it relates to people, does it unfairly advantage or disadvantage a particular social group? No.

Context:
Mr. Bingley, a wealthy man, leases Netherfield Park and is well-liked by everyone. Jane is attracted to him, and Elizabeth thinks he might feel the same way about her. Darcy begins to notice Elizabeth's intelligence and starts to pay attention to her.

Current Segment:
Mr. Collins, a clergyman, arrives to court one of the Bennet girls. He decides to choose Elizabeth, and after a conversation with Mrs. Bennet, he changes his mind to Jane.

✓ STEP 1: Highlight Error Span (from Current Segment)
he changes his mind to Jane

STEP 2: Choose Error Type

New Character w/o Introduction Missing Reference to Entity/Event Unclear Coreference

Abrupt Transition from Previous Scene Repetition Inconsistent Ungrammatical/Nonsensical

Previous sentence Coherent / No-more errors. Go to next segment.

Submit

Tutorial tldr;

Steps for annotating an Error:

- Step 1: Highlight Error Span
- Step 2: Select Error Type (Definitions given below)
- Step 2.5: For Repetition and Inconsistent Errors, select earlier span.
- Step 3: (Optional) Give feedback
- Step 4: Click on Add

Error Type Definitions

(click on type for definition)

- New Person not Introduced
- Missing Information about an Event/Object
- Abrupt Transition from the Previous Scene
- Unclear Coreference
- Repetition
- Inconsistent
- Ungrammatical/Nonsensical Text

PREVIOUS ANNOTATIONS:

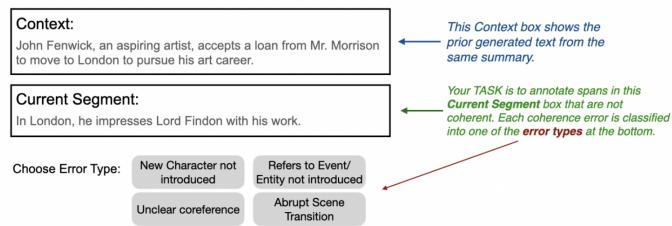
Remove	Abrupt Scene Transition -- Spans: ... feel the same way about her. Darcy begins to notice Elizabeth's intelligence and starts to pay attention to her. ..., Segment: 0
Remove	New Person not Introduced -- Spans: ... feel the same way about her. Darcy begins to notice Elizabeth's ..., Segment: 0
Remove	New Person not Introduced -- Spans: ...Jane is attracted to him, and Elizabeth thinks he might feel the same..., Segment: 0
Remove	New Person not Introduced -- Spans: ...nd is well-liked by everyone. Jane is attracted to him, and Eliz..., Segment: 0

Figure 16: Screenshot of the task interface for SNAC annotations

Basic Task Description Next

Thank you for participating in this study! The goal of this study is to **read machine-generated summaries of books or news articles and identify coherence errors in the text.** These are errors where individual sentences in a summary might make sense, but they don't fit together, like if a sentence talks about two people fighting and then the next sentence talks about someone totally unrelated having tea. We're going to guide you through the error types you'll be annotating.

You will see a display like the one pictured below. **Your task is to annotate coherence errors in the text shown in the Current Segment box.** The Context box shows earlier generated text from the same summary. Identifying a coherence error will generally involve picking a type of error, which we'll cover later, and highlighting the span of text containing that error.



In one HIT, you will be asked to annotate coherence errors for each text segment in a summary. At the start of the task, you will annotate the first text segment and the context box will be empty. Subsequently, you will be asked to annotate errors in the following text segments. The Context and the Current Segment boxes will update their contents accordingly.

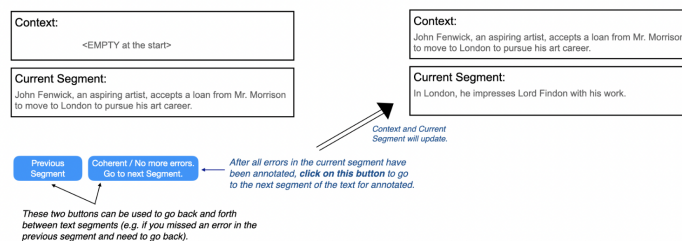


Figure 17: Screenshot of the first page of the tutorial provided to crowd annotators

Next

Previous

Annotation Workflow

Next, we will describe the steps to annotate coherence errors in the **Current Segment** text. Note that a single segment may contain multiple coherence errors (often with overlapping text spans). Each of these should be annotated independently.

To annotate an error (also demonstrated in the image below):

- Step 1: Highlight the Span from Current Segment that contains the error. The highlighted span will automatically get populated in the input field.
- Step 2: Choose the Error Type
- Step 3: (Optional) Provide more feedback/comments for the annotation. You can use this text box to indicate if you are unsure about this categorization, or to explain your selection if required. For a majority of the annotations, we expect this field to be left blank.
- Step 4: Click on the **Add** button.

Once you've annotated **ALL** errors in a span, click on the **Coherent / No more errors. Go to next segment** button to proceed to the next segment for annotation.

Once all text segments in the summary are annotated, the **Submit button will appear. Click on this to submit the HIT and generate the Mechanical Turk code.**

Note: All annotated errors are shown in a table at the bottom of the page. You can delete previously annotated errors.

Previous Annotations

Remove New Character not introduced: he impresses Lord Findon with...

Click on **Remove** to delete a previous annotation.

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

Previous

End of the Tutorial and Example Annotation

This is the end of the tutorial! You can now go back to the HIT and complete the task.

For reference, here is an example of how your annotation for a given **Current Segment** may look like. Read the Context and the Current Segment text, then Click on 'See Annotations' to see our annotations. (Hint: there are around 3 different errors).

Context:
Jonathan arrives in Bistriz and is greeted by Count Dracula who insists on carrying his luggage. Jonathan realizes he's a prisoner and resolves to watch the Count carefully.
Lucy receives multiple marriage proposals but politely declines them as she already has feelings for Jonathan.

Current Segment:
Mina wakes up to find Lucy trying to get out of the room multiple times during the night. Lucy's wounds on her neck have not healed and Mina fears they may become infected.

Show Annotations

New Person not introduced: The text refers to 'Mina' who is a new character not introduced in the previous story.

Abrupt transition from previous scene: The setting of the story has changed abruptly.

Note: Lucy is also not introduced in the story, but we only annotate errors once, and therefore 'Lucy' should be annotated in the previous text segment.

Note: Sometimes these can be fixed by adding discourse connectors like 'One day'.

Mina wakes up to find Lucy trying to get out of the room multiple times during the night.
Lucy's wounds on her neck have not healed and Mina fears they may become infected.

Missing reference to entity/event: The earlier context does not refer to any injury or wound. This is new information that is relevant to understand the narrative.

Figure 19: Screenshot of the last page of the tutorial provided to crowd annotators