Jigsaw Pieces of Meaning: Modeling Discourse Coherence with Informed Negative Sample Synthesis

Shubhankar Singh
Mercer Mettl
shubhankar.singh@mercer.com

Abstract

Coherence in discourse is fundamental for comprehension and perception. Much research on coherence modeling has focused on better model architectures and training setups optimizing on the permuted document task, where random permutations of a coherent document are considered incoherent. However, there's very limited work on creating "informed" synthetic incoherent samples that better represent or mimic incoherence. We source a diverse positive corpus for local coherence and propose six rule-based methods leveraging information from Constituency trees, Part-of-speech, semantic overlap and more, for "informed" negative sample synthesis for better representation of incoherence. We keep a straightforward training setup for local coherence modeling by fine-tuning popular transformer models, and aggregate local scores for global coherence. We evaluate on a battery of independent downstream tasks to assess the impact of improved negative sample quality. We assert that a step towards optimality for coherence modeling requires better negative sample synthesis in tandem with model improvements.

1 Introduction and Motivation

Coherence is the bridge between elements of discourse which imposes strong logical connections, semantic relationships, smooth transitions, and thematic progressions. Halliday and Hasan (1976) formally defined coherence as a text's interpretive unity through cohesion, introducing Local and Global Coherence concepts, the former addressing connections between adjacent text units, while the latter looking at the broader discourse organization for a document. van Dijk (1977) additionally emphasizes the role of macrostructures and cognitive processes, going beyond mere textual properties. Coherence modeling has been a fundamental task in discourse and pragmatics, with applications in text generation, dialogue systems, and reasoning, yet presents formidable challenges in modeling and a veritable lack of quality data.

Entity-based models (Barzilay and Lapata, 2008; Elsner and Charniak, 2011) capture patterns of entity distribution in text by focusing on the roles of salient entities (Grosz et al., 1995). To this, Tien Nguyen and Joty (2017) apply a neural approach using convnets. Rhetorical Structure Theory (RST) based methods formalize coherence as discourse relations (Louis and Nenkova, 2012; Mann and Thompson, 1988). Li and Hovy (2014) feature recurrent layers to encode individual sentences within 3-sentence windows. Li and Jurafsky (2017) use an encoder-decoder architecture to incorporate global topic information. Mesgar and Strube (2018) model changes in salient semantic information. The transferable Neural model (Xu et al., 2019) focuses on local coherence, training forward and backward models on adjacent sentences, along with generative pre-training of sentence encoders. The Unified Coherence model, proposed by Moon et al. (2019), is highly regarded for its impressive results, employing a Siamese framework with a bilinear layer and lightweight convolution pooling.

Coherence models often learn and evaluate using a pairwise-ranking task on the Wall Street Journal (WSJ) Corpus Documents. An original document serves as a coherent "positive" sample, while its permuted version is the incoherent "negative" sample. The primary goal is to train models to predict a higher coherence score for the original than its random permutations and determine total accuracy. Introduced by Barzilay and Lapata (2008), the corpus and task have been prime sets for most research in modeling and evaluating coherence. Mohiuddin et al. (2021) assessed state-of-the-art models trained on the WSJ permuted data. While the models excelled in the permuted document task, they struggled in downstream evaluations. Pishdad et al. (2020) note that success on the permuted document task doesn't fully reflect true coherence modeling

Findings of the Association for Computational Linguistics: EACL 2024, pages 1895–1908

abilities advocating for broader evaluations of these models.

Jwalapuram et al. (2022) present the state-of-the-art for the pairwise WSJ task using an extensive contrastive setup that contrasts positive samples with permuted negatives via automatic hard negative mining to harness "harder" samples during training. This approach, leveraging hard-mining negative samples during training, achieves improved results. Shen et al. (2021) adopted a different approach from random permutations, focusing on intruder-detection. To formulate incoherent documents using the CNN and Wikipedia corpora, they substitute a sentence from a coherent document with a comparable sentence from a different document. Through bigram hashing and TF-IDF matching, they retrieve 10 similar documents, then choose a random non-opening sentence from these to create 10 potential replacements. They further refine the substitution using filters based on TF-IDF similarity, thereby making an "informed" change that turns a positive document into a negative one. Their findings indicate that fine-tuned transformer models excel at this task.

Based on this we propose that relatively straightforward training setups akin to document classification using pre-trained models and aggregation can yield comparative or better scores against prominent models for coherence, achieved by creating more *sophisticated "informed" synthetic samples for incoherent data leveraging granular and nuanced syntactic and semantic text information*, as opposed to the simpler data curations based on random permutations that many complex models and setups currently rely on.

Incoherent "negative" samples from six, rule-based-heuristic, "informed" negative data synthesis processes are crafted from a novel 3-sentence locally coherent "positive" text corpora obtained from diverse sources after a curated extraction process. These 3-sentence local windows are used to fine-tune transformer models, from which a simple aggregation method yields a global document coherence estimation system. This system is then evaluated on a battery of downstream evaluations and compared against prominent models.

We achieve results comparable to state-of-the-art models trained explicitly on the WSJ permutation training set, with fast convergence and significantly better performance on a logical coherence evaluation test. We conduct an ablation analysis examining the incoherent sample synthesis methods, followed by a discussion. Our conclusion emphasizes the importance of nuanced incoherent data synthesis that mimics natural incoherence. Scripts made available[1] (refer ethics statement).

SRC	Samples	AWC	VS
WKI	54,991	67.95	97,278
ROC	59,890	30.04	19,149
ARX	27,228	70.89	31,197
BKP	12,258	64.82	38,288

Table 1: Positive Summary: The number of samples, average word count per window, and vocabulary size for the windows in each set.

2 Extracting Coherent Samples

We select a 3-sentence window for our local coherence analysis (Li and Hovy, 2014; Moon et al., 2022). Our locally coherent "positive" set is curated after an extraction and filtration process from four diverse sources of text: **Arxiv Abstracts** - **ARX** - Summaries of academic literature, **Wikipedia "Good"** - **WKI** - Articles tagged to be "good" on Wikipedia [2], **ROC Stories** - **ROC** - Short commonsense stories (Mostafazadeh et al., 2016), **Book Plots** - **BKP** - Book plot texts [3]. For ROC we eliminate all samples that may have any overlap with the StoryCloze test which we evaluate on later (1571 samples). Text from all sources is human-written.

We iterate over and parse documents from each source into lists of sentences using a parser except for ROC where sentences are pre-parsed. From these sentence lists, we extract three-sentence windows. Every sentence undergoes cleaning to remove unicode errors and filter URLs/tags. Moreover, as an additional filtration heuristic, each sentence is evaluated for linguistic acceptability using the 'textattack/roberta-base-CoLA' model (Morris et al., 2020) trained on COLA (Warstadt et al., 2019). If a sentence in a window fails the check, the window is discarded. On average, 5.21% of windows per set are rejected. We ensure significant distance and no overlaps between windows from the same document. The detailed extraction process is explained in Algorithm 1. The summary of the positive corpus is presented in Table 1.

[1] github.com/shubh11220/Coherence (refer ethics)
[2] en.wikipedia.org/wiki/Wikipedia:Good_articles
[3] kaggle.com/datasets/athu1105/book-genre-prediction

Algorithm 1 Window Extraction

Require: Source Files F_{SRC}
Ensure: All other functions are defined
for f in F_{SRC} **do**
  for doc in f **do**
    $sents \leftarrow Parser(doc)$
    if $len(sents) < 3$ **then** `continue`
    end if
    Split $sents$ to $groups$ (2 < LEN < 7)
    for each G in $groups$ **do**
      $L_g \leftarrow len(G)$
      $i \leftarrow random(0, L_g - 3)$
      $w \leftarrow [G[i], G[i+1], G[i+2]]$
      for sen in w **do**
        $C \leftarrow Clean(sen)$
        if not Acceptability(C) **then**
          `continue to next group`
        end if
      end for
      Add w to $Windows$
    end for
  end for
  Store $Windows$ in a DataFrame and save
end for
Ensure: Individual source sets saved at F_{DEST}

3 Negative Samples

We craft incoherent samples using six methods to perturb samples directly from the positive set, ensuring positive-negative samples remain within the same general space.

M1 and ***M2*** incorporate syntactic details from sentences to execute informed substitutions. They primarily focus on modifying the contextual and descriptive elements of the sentences:

M1. Constituency Subtree Substitution: Subtree substitution has been an explored topic in the NLP predicament especially for data augmentation (Shi et al., 2021; Yang et al., 2022). We substitute Prepositional Phrases (PP), Adjective Phrases (ADJP) and Adverb Phrases (ADVP) in positive sample sentences. By replacing the ADJP, ADVP, or PP modifiers, we change the "Where," "How," and "Why" of a sentence, not the "Who" or "What".

Using a neural constituency parser (Kitaev and Klein, 2018), we flatten the positive corpus, extract a subset, iterate over sentences, and form a dictionary of ADJPs, ADVPs, and PPs called *Bank* (B). For a given sentence S and B with keys: $ADJP, ADVP$, and PP, if constituency parse tree structure S contains subtree with $key \in \{ADJP, ADVP, PP\}$, it generates a set of 5 candidate replacements $S'_{candidates} = \{S_1, S_2, \ldots, S_5\}$, where each S_i is a variant of S with the key text substituted from $B[key]$. The candidate S' is chosen such that $S' = argmax_i(Acceptability(S_i))$ (Acceptability is modeled similarly to the positive method). This process is applied to a maximum of two sentences in each positive window W, with the number of substitutions constrained by $1 \leq \text{substitutions} \leq 2$. A visual depiction is provided in Figure 1(a).

M2. Salient Token Substitution: A method to model entity-based incoherencies. Draws parallels with the prior method. We identify contextually salient Part-Of-Speech (POS) Tags that are linked to salient tokens in the sentence, specifically nouns, verbs, and adjectives $\mathcal{L} = \{NN, NNS, NNP, NNPS, VB, VBD, VBG, VBN, PRP, JJ, JJR, JJS\}$. These tags convey salient information regarding the sentence's entities and their interrelations. Analogous to ***MI***, we construct a *Bank* B by flattening the positive corpus, parsing, and mapping POS tags to token replacement lists. From the positive window, a **single sentence** is chosen at random, parsed, and tokens bearing these vital POS tags are identified and appended to a salient token set. On randomly discarding 70% of these tokens from the set, the remaining 30% are substituted in the sentence using dictionary tokens having an identical tag. We discard 70% tokens to so as to not drastically perturb the sample. The sentence is reinserted into the window. This is done for each window in every positive source set. We choose the top 35% linguistically acceptable windows at the end. Contrasting with ***M1***, this methodology introduces incoherencies concerning correctness as well. A visual illustration of this method can be seen in Figure 1(b).

M3 and ***M4*** are intruder sentence injection methods, selecting a sentence from a positive sample for substitution based on similarity and saliency heuristics. ***M3*** and ***M4*** flatten **each** source set in the positive corpus **individually** to bags of sentences to select intruder sentences. Both iterate on each window substituting a single sentence.

M3. Similarity Intruder Injection: Given a positive source set P, for each window W in P, we first select 12 candidate intruder sentences $I_{candidates} = \{I_1, \ldots, I_{12}\}$ at random from P's corresponding *bag* B_{P}, where B_{P} is a flattened list

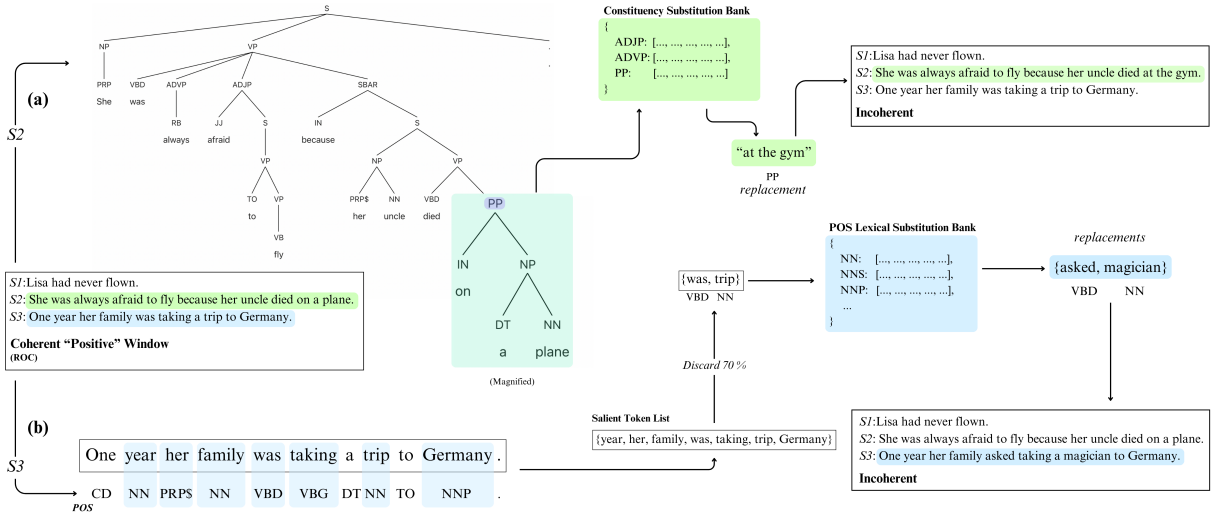

Figure 1: A rough overview of ***M1*** and ***M2*** pipelines visualised.

of all sentences in all windows in P. For each candidate I_j, the cosine similarity $cos(I_j, W)$ is computed against the entire window's document embeddings using Sentence Transformers (Reimers and Gurevych, 2019). We select the sentence I^* such that: $I^* = argmax_j(cos(I_j, W))$. The selected intruder I^* is then used to substitute any one of the three sentences in the window W, provided: $cos(I^*, W) \geq 0.2$. An observational grid-search-like process determined this minimum threshold and the parameter of twelve candidate replacements. These parameters ensure that the intruder sentence maintains some similarity with the window while preserving a degree of randomness to ensure incoherence.

M4. Token Overlap Intruder Injection: Let $\mathcal{L}$ denote the salient Part-Of-Speech (POS) list from ***M2***. In ***M4***, we enhance $\mathcal{L}$ to also include pronouns: $\mathcal{L}_{M4} = \mathcal{L} \cup \{PRP\$, WP, WP\$, WRB\}$. For a given positive source set P each window W in P has a saliency representation of tokens $S_W = \left[\{t_1^1, \ldots, t_m^1\}, \{t_1^2, \ldots, t_n^2\}, \{t_1^3, \ldots, t_o^3\}\right]$. The flattened window saliency set, $F_W = \{...\}$, accumulates the saliency sets from its sentences. Each token in every saliency set is lemmatized. We construct a bag B_P per source set, like ***M3***, containing linked saliency sets for each sentence. We define a selection value W in P as $num = \text{len}(P) \times 0.1$. For each W, after selecting num random sentences from B_P and obtaining F_W, the overlap between F_W and all candidate replacement saliency sets in B_P is calculated. The overlap for a candidate set C is denoted by $Overlap(F_W, C)$ with the chosen candidate replacement, C^*, satisfying $C^* = \text{argmax}_C(Overlap(F_W, C))$ constrained within $0.3 \leq Overlap(F_W, C^*) \leq 0.6$. These constraint and selection values are derived from observational analysis like in ***M3***. Ultimately, C^* substitutes a random sentence in W.

M5 and ***M6*** serve as supplementary methods, introducing incoherencies related to the correctness and structural integrity of sentences. While these aspects may not be paramount in broader discourse, they can be integral on a more granular level. For a given positive source set P with each window W in P we apply them to a single sentence S in the window. For both ***M5*** and ***M6*** we construct the saliency set for S like in ***M4***:

M5. Intra-Sentence Permutation Like in ***M2*** we shorten this set by randomly discarding 70% of total tokens. The remaining tokens in the set are permuted for their positions with each other in the sentence.

M6. Context Dissipation Unlike ***M5*** we do not permute the 30% set tokens from the sentence but simply delete them.

The final summary of negative samples is presented in Table 2. Our methodology for generating negative samples aimed for a theoretical maximum of six negatives per positive instance, utilizing methods ***M1*** through ***M6***. The actual yield was moderated by the application of thresholds and heuristic cutoffs, particularly in ***M3*** and ***M4***, to preclude drastically perturbed samples, alongside linguistic acceptability criteria in ***M1***, ***M2***, ***M5***, and ***M6***. The resultant ratio represents the viable negatives effectively utilized. **Examples for these are present in the Appendix section of the paper.**

Method	Samples
M1.	52,255
M2.	60,834
M3.	61,178
M4.	39,943
M5.	15,906
M6.	10,091

Table 2: Negative Sets Summary

System	Acc.	Prec.	Rec.	F_1
BERT base $_{\text{No FT}}$	77.5	72.5	81.7	76.8
BERT base	89.8	81.3	93.5	87.0
BERT large	91.9	83.9	95.0	89.1
DistilBERT	91.0	84.1	93.9	88.7
XLNET base	90.3	82.8	94.8	88.4
XLNET large	92.5	86.8	95.1	90.8
RoBERTa base	92.1	85.7	94.7	90.0
RoBERTa large	93.5	88.5	95.8	92.1

Table 3: Test Accuracy, Precision, Recall and F_1 score.

4 Coherence Modeling

Our main model is the local coherence model which is based on a fairly straightforward fine-tuning setup. The global document coherence modeling (DCM) setup is based on the local model itself.

4.1 Local Coherence Model

Local coherence modeling is framed as a binary classification task. A model takes in 3-sentence text windows and predicts a score. This method bears resemblance to BERT's Next Sentence Prediction (NSP) task (Devlin et al., 2019), the difference primarily being the type of sentences and the context length. We by fine-tuning prominent transformer-based encoder models such as BERT (2019), DistilBERT (2019), XLNet (2019), RoBERTa (2019) (and their large versions).

For a window W comprising 3 sentences *(sen1, sen2, sen3)* (whitespace separated), our model leverages representations from BERT-based encoders (characterized by ϕ) to determine a coherence score for the sentences together as a document separated by white spaces. Specifically, for a document d_i having k tokens $(w_1, w_2, \ldots, w_k)$, transformer encoder models transform each token w_t into its vector form $v_t \in \mathbb{R}^d$, where d signifies the embedding's dimension. Additionally, the entire input D is converted into a document vector $z \in \mathbb{R}^d$, representing the [CLS] token. A linear layer is then appended to transform this document vector z, producing the final coherence score: $f_\theta(D) = w^\top z + b$. Here, w and b represent the weight and bias of the added linear layer.

4.2 Document Coherence Modeling Setup

For our global, document coherence setup, we target documents in a 4 to 10 sentence range. This aligns with prevailing research practices, where the segment of a document under consideration typically reflects a paragraph or a section with up to 10 sentences. For larger documents, segmentation may be required.

Given a document D of length n, our approach employs the local coherence model to infer a global coherence score. This score is conceived as a mean of the local coherence scores found within the document. To decompose the document structure, we employ a sliding window mechanism, using a 3-sentence context window that moves from the beginning of the document with a single stride, while abstaining from any padding. This approach results in $n-2$ windows for the given document length.

To these windows we additionally incorporate one-hop windows (which augment our data and capture information at a distance) from the document where the window consists of sentences at i, $i+2$, and $i+4$. We obtain all within-range one-hop windows. Thus, our total set of windows encompasses no-hop and 1-hop windows (Although, we noticed only marginal improvements after including the 1-hop windows in the downstream tasks). Using the local coherence model, we compute the local coherence scores for all these windows. The final score S_g for the document D is the mean of window scores.

We maintain this setup to be straightforward and clear to ensure that any comparisons in our performance on downstream evaluations are largely attributed to the quality of our corpora, rather than innovations in model architecture or training setups. We aim to evaluate how our strategy, which emphasizes diverse positive data and curated "informed" negative samples, compares to the more complex state-of-the-art models and training setups.

4.3 Training

We compile our dataset from positive (154K samples) and negative sets (240K samples, detailed in

Table 2), resulting in around 394K samples split into train, test and dev sets at a 70/20/10 ratio. Consistent fine-tuning hyperparameters are used across pre-trained models with a dropout rate of 0.2 on the base model and linear layer, and a reduced max length. Training spans 3 epochs with the AdamW optimizer (Loshchilov and Hutter, 2019), with a linearly decreasing rate scheduler with Binary Cross-Entropy (BCE) Loss. We train on Nvidia A100 GPU instances. Inference metrics like accuracy, precision, recall, and F_1 score from the test set are in Table 3. The results reported are a mean of 5 runs.

We observe that the RoBERTa-large model performs the best for all metrics and we use the XLNet large and RoBERTa large variants for our document coherence modeling (DCM) setup for further downstream evaluation. We record lower precision scores than recall for most of our models, which is informative as it tells us that our negative samples are sufficiently hard which are then being classified as positive.

5 Downstream Evaluations

We test our document coherence modeling (DCM) approach on a battery of downstream task-independent pairwise test sets similar to Jwalapuram et al. (2022). These include the WSJ Test Set, SummEval Annotated Set (Fabbri et al., 2021), INSteD-CNN - INSteD-Wiki Sets (Shen et al., 2021) and the StoryCloze Test (Mostafazadeh et al., 2016).

We use a pairwise setup where the score of a positive sample is ranked against a negative one, measuring on total accuracy. Pairwise comparisons are scale-invariant, they focus on relative score positions thus, despite varied task or dataset scales, the evaluation is consistent. We also test on the GCDC test sets (Lai and Tetreault, 2018) for pairwise ranking and minority class prediction to compare with benchmarks and assess natural use cases.

We compare against state-of-the-art baseline models with previously reported scores on these tasks: Local Coherence Discriminator (**LCD**) model (Xu et al., 2019): (i) **LCD-G** with GloVe representations (Pennington et al., 2014), (ii) **LCD-I** using InferSent (Conneau et al., 2017), and **LCD-L** from an RNN-trained language model; (**UNC**) model (Moon et al., 2019) and the **Contrastive** and Contrastive with Hard-Mined Negatives (**HMN**) model (Jwalapuram et al., 2022). For GCDC we have the **LEXGRAPH** (Barzilay and Lapata, 2008), **EGRAPH** (Guinaudeau and Strube, 2013), **CLIQUE** (Li and Jurafsky, 2017) and **SENTAVG**, **SENTSEQ/PARSEQ** models from Lai and Tetreault. All these prominent models allow for a good comparison as they have exhibited excellent results on a myriad of downstream sets in the past.

5.1 Tasks

WSJ: Benchmark for global coherence tasks contrasts a document against 20 of its random sentence permutations, excluding any matching the original. Documents undergo 20 permutations in a pairwise test, comparing coherence scores. Testing uses Moon et al. (2019)'s set with 20,411 pairs from 1053 documents (Sections 14-24 of the WSJ corpus).

SummEval: The SummEval collection of human judgments of model generated summaries on the CNN Dailymail dataset (Fabbri et al., 2021) consists 1600 model generated summaries by 16 generation systems on 100 articles (Chen et al., 2016). Each summary has coherence ratings from three expert annotators using a Likert-like scale. Jwalapuram et al. (2022) adapts this to a pairwise setup pairing summaries for every system and unique source article. The summary with superior coherence becomes the positive document, while its counterpart is the negative one. This yields $\binom{16}{2} \times 100 = 12,000$ pairs for assessment. A constraint to consider is the notably low inter-annotator agreement (Krippendorff's alpha - 0.492 For workers, 0.413 for experts, improved to 0.712).

Story Cloze Test: This is an independent commonsense reasoning set proposed. Following on Pishdad et al. (2020), we assess models using the StoryCloze dataset (Mostafazadeh et al., 2016). This dataset offers short narratives with two endings, one being implausible and logically incoherent. Using the validation set (as test labels are private), we pair narratives with correct endings as positive and incorrect ones as negative, yielding 1,571 evaluation pairs. As outlined in section 2, any windows that contained even a single sentence from these test samples were removed from our ROC set prior to training.

INSteD: As introduced previously, the task presented by Shen et al. (2021) to assess the coherence abilities of pre-trained language models by detecting intruding sentences is again adapted to a pairwise setting. The pairwise framework pairs the original document with its corrupted incoherent

System	SummEval	StoryCloze	INSteD-CNN	INSteD-Wiki	WSJ
LCD-G	54.15±0.83	51.76±1.22	61.24±0.71	55.09±0.46	90.39±0.28
LCD-I	51.71±0.99	52.69±0.69	60.23±0.86	53.50±0.37	91.56±0.16
LCD-L	53.56±1.20	50.09±1.57	55.07±0.26	51.04±0.47	90.24±0.36
UNC	46.28±0.80	49.39±1.81	67.21±0.55	55.97±0.45	94.11±0.29
Contrastive	66.93±1.10	72.83±2.89	92.84±0.61	71.86±0.69	98.59±0.20
Contrastive-HMN	67.19±0.63*	74.62±2.79	93.36±0.49*	72.04±1.05*	98.58±0.18*
XLNet-large-DCM	61.89±1.20	76.32±1.37	91.11±0.61	70.16±0.65	92.42±0.53
RoBERTa-large-DCM	62.45±1.17	77.42±1.81*	92.32±0.28	71.33±0.87	93.79±0.41

Table 4: Results (net pairwise-accuracy on various independent evaluations. All models except for ours are trained explicitly on the WSJ permute task. Results are a mean of 5 runs. {*} Represents the top scores. All models except for ours are trained explicitly on the WSJ data as detailed in Jwalapuram et al. (2022)

System	Yahoo	Clinton	Enron	Yelp
EGRAPH	**64.0**	75.3	75.9	59.5
LEXGRAPH	62.5	78.3	77.9	60.8
CLIQUE	57.8	89.4	88.7	64.6
SENTSEQ	58.3	88.0	87.1	**74.2**
XLNet-lg.-DCM	62.7	89.1	86.9	72.1
RoBERTa-lg.-DCM	63.8	**90.2**	**89.4**	73.3

Table 5: Pairwise Sentence ordering accuracy on GCDC test sets. The top score is highlighted for each set.

System	Yahoo	Clinton	Enron	Yelp
EGRAPH	0.308	**0.382**	0.278	0.117
CLIQUE	0.055	0.000	0.077	0.146
SENTAVG	**0.481**	0.332	**0.393**	0.199
PARSEQ	0.447	0.296	0.373	0.112
XLNet-lg.-DCM	0.431	0.310	0.374	0.194
RoBERTa-lg.-DCM	0.462	0.336	0.384	**0.211**

Table 6: Minority class predictions, $F_{0.5}$ score on GCDC test sets. The top score is highlighted for each set.

counterpart. This provides 7,168 pairs from their CNN test set (INSteD-CNN) and 3,666 from the Wikipedia set (INSteD-WIKI) for evaluation.

GCDC: Lai and Tetreault (2018) provide a real-world text corpus to model coherence, the Grammarly Corpus of Discourse Coherence (GCDC), incorporating texts from the Yahoo Answers L6 Corpus, Clinton & Enron Mails Corpora, and the Yelp Open Dataset, with 200 test samples from each source. Our evaluation delves into two primary tests of this dataset: sentence ordering (pairwise setting) and minority class prediction. The former follows a setting similar to the WSJ evaluation (20 random permutations), specifically targeting texts with high coherence (gold rating 3). For sets Yahoo, Clinton, Enron and Yelp containing 76, 111, 88 and 108 positive samples respectively we get a total of 7660 test samples. The minority class prediction aims to categorize a subset where only 15-20% is labeled as low coherence. Texts are designated "low coherence". The $F_{0.5}$ score, which favors precision over recall serves as the evaluation metric. Echoing the patterns in SummEval annotations, there's a discernible low inter-annotator agreement across these datasets: Mean Intra-Class Correlation coeff. (ICC) for experts for all sets being 0.422.

5.2 Results

Results for the pairwise independent sets are presented in Table 4. Tables 5 and 6 present results for the GCDC test sets.

In the independent pairwise tests, both our setups, XLNet-large-DCM and RoBERTa-large-DCM (DCM: Doucment Coherence Modeling), notably outperformed the non-contrastive models (LCD-G, LCD-I, LCD-L, and UNC) across all evaluation tasks. When compared with contrastive models, our models exhibited competitive performance. Specifically, our approaches closely matched the highest scores, with a notably higher performance in the StoryCloze test aimed at detecting incoherencies in logical and narrative flow, where they surpassed others by a significant margin. In other tasks, our models showed close performance to the Contrastive and Contrastive-HMN models, with the margin being relatively small. This is a significant result, emphasizing the capability of our models to

Removed	Acc.	SE	Cloze	IN-CNN
None	90.9	55.8	71.6	83.4
M1, M2	92.8	55.2	66.3	81.2
M3, M4	93.4	54.8	67.2	80.6
M5, M6	90.1	52.4	72.3	83.1

Table 7: Ablation results (net pairwise-accuracy) on various independent downstream evaluations.

perform on par with state-of-the-art models. We didn't achieve a comparable score for the WSJ task, largely because other models were specifically trained on the WSJ train set. For the GCDC sentence ordering tasks, we are able to outperform the others on the Clinton and Enron sets. Similarly, on the minority class prediction task we outperform on the Yelp set. On all the other sets for both the tasks our results are competitive.

Our results are well distributed, competitive and go on to show that better quality data in termed of diversity and "informed" negative samples for the task, is a parallel facet of this research.

5.3 Ablation Analysis

We carry out a restricted ablation analysis to address two primary questions: 1. Among the methods of generating negative samples, which are "easier" for a model to grasp? 2. How do these methods influence specific independent tasks? Our approach involves randomly selecting 80K positive samples and 120K negative samples, ensuring a higher number of negatives. From the complete set of negative samples, we exclude pairs of related sets, specifically **[M1, M2]**, **[M3, M4]**, and **[M5, M6]**, and then select the 120K samples. We then fine-tune the RoBERTa-base model on this collective 220K sample set with consistent conditions. We use downgraded settings and model for better distinction in our study. We set a baseline for these settings in which we don't remove any negative set. We evaluate on test accuracy (within the training samples) and pairwise SummEval (SE), StoryCloze (Cloze) and INSteD-CNN (I-CNN) downstream sets.

We report the results in Table 7. In response to our first question, we noted the test accuracy is lowest when **[M5, M6]** are removed, and it's higher when other methods are excluded, given the prevalence of **M5**, **M6** samples in the 120k quota when other sets are removed. Thus incoherencies related to structure and correctness are the easiest for a model to grasp. On the contrary, when we remove **[M1, M2]** or **[M3, M4]** we observe that the test accuracy goes up indicating they are indeed 'harder' samples when compared to **M5** and **M6**.

We noticed that removing **M5** and **M6** causes the most significant drop in SummEval accuracy. StoryCloze's accuracy diminishes with the exclusion of **[M1, M2]** and **[M3, M4]**, but less so when **[M5, M6]** are removed, suggesting the first four methods mainly influence logic-based incoherencies. INSteD-CNN's value drops most notably without **[M3, M4]**, with a comparable decrease when **[M1, M2]** are excluded. Overall, informed negative samples significantly impact results.

6 Conclusion and Future Work

In this paper, we take a parallel approach to coherence modeling as opposed to optimization on the permuted document task by sourcing a diverse positive corpus and synthesizing "informed" incoherent samples from the positive corpus with six methods utilising constituency parse information, POS, semantic similarity and more. We perform local coherence model training using a simple fine-tuning setup and form a score aggregation method for global document coherence modeling. Using this setup we test on multiple independent downstream tasks which capture some form on incoherence in the text. Our nuanced approach to forming negative samples and obtaining scores results in getting comparable performance in the tasks (particularly standing out in a few) against many popular models and training setups developed for this task. The efficacy of our models in diverse evaluations, along with our findings, highlights the pivotal role of sophisticated, "informed" negative sample synthesis in advancing the field of coherence modeling. In the future, we plan to expand our scope by training more curated models on this training data such as contrastive models, siamese networks, and more. While these methods are designed to be domain-agnostic, there is an interest in exploring the nuances of incoherence within specific, context-rich discourse domains, such as the medical or legal fields, effectively investigating domain-specific incoherence. We're interested in exploring how generative techniques, such as GANs or human-in-the-loop systems, can aid in producing incoherent samples and assist in mining hard negatives during the incoherent text generation phase. A multilingual angle for this can also be explored.

Limitations

We aim to address several limitations in our future work. Firstly, the inherent limitations or biases in pre-trained transformers can influence the outcomes, and alternative architectures might be better suited for the task. Secondly, our described training setup, although straightforward, might not be robust enough to address intricate incoherence or capture nuances present in more complex training environments. Lastly, while the insights from our ablation analysis are valuable, they may not be exhaustive, and there might be unidentified underlying factors impacting performance. We do not propose a direct training model but methods that may improve modeling on the task. There may be more such linguistically grounded methods to craft negative samples which must be explored.

Ethics Statement

Adhering to ethical standards, particularly with data sources (both positive source and downstream evaluation sets) requiring permissions, we provide scripts and partial data rather than full datasets, emphasizing our commitment to responsible data sharing and practical application within ethical guidelines. Our methods, versatile and multilingual, apply to various text types and extend to tasks like dialogue response generation. Additionally, some models and scripts are designed for **potential production use in our own proprietary text evaluation systems**.

Acknowledgements

We would like to thank the anonymous reviewers of EMNLP 2023 and EACL 2024 (ACL ARR) who have helped improve this work. We also sincerely thank the owners/creators for the source and downstream evaluation sets for helping us conduct this work.

References

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

Danqi Chen, Jason Bolton, and Christopher D. Manning. 2016. A thorough examination of the CNN/Daily Mail reading comprehension task. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2358–2367, Berlin, Germany. Association for Computational Linguistics.

Alexis Conneau, Douwe Kiela, Holger Schwenk, Loïc Barrault, and Antoine Bordes. 2017. Supervised learning of universal sentence representations from natural language inference data. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 670–680, Copenhagen, Denmark. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

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, Portland, Oregon, USA. Association for Computational Linguistics.

Alexander R. Fabbri, Wojciech Kryściński, Bryan McCann, Caiming Xiong, Richard Socher, and Dragomir Radev. 2021. Summeval: Re-evaluating summarization evaluation.

Barbara J. Grosz, Aravind K. Joshi, and Scott Weinstein. 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, Sofia, Bulgaria. Association for Computational Linguistics.

M Halliday and Ruqaiya Hasan. 1976. *Cohesion in English*. Longman Group Limited, London, UK.

Prathyusha Jwalapuram, Shafiq Joty, and Xiang Lin. 2022. Rethinking self-supervision objectives for generalizable coherence modeling. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 6044–6059, Dublin, Ireland. Association for Computational Linguistics.

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

Alice Lai and Joel Tetreault. 2018. Discourse coherence in the wild: A dataset, evaluation and methods. In *Proceedings of the 19th Annual SIGdial Meeting on Discourse and Dialogue*, pages 214–223, Melbourne, Australia. Association for Computational Linguistics.

Jiwei Li and Eduard Hovy. 2014. A model of coherence based on distributed sentence representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 2039–2048, Doha, Qatar. Association for Computational Linguistics.

Jiwei Li and Dan Jurafsky. 2017. Neural net models of open-domain discourse coherence. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 198–209, Copenhagen, Denmark. Association for Computational Linguistics.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized BERT pretraining approach. *CoRR*, abs/1907.11692.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization.

Annie Louis and Ani Nenkova. 2012. A coherence model based on syntactic patterns. In *Proceedings of the 2012 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning*, pages 1157–1168, Jeju Island, Korea. Association for Computational Linguistics.

William C. Mann and Sandra A. Thompson. 1988. Rhetorical structure theory: Toward a functional theory of text organization. *Text & Talk*, 8:243 – 281.

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, Brussels, Belgium. Association for Computational Linguistics.

Tasnim Mohiuddin, Prathyusha Jwalapuram, Xiang Lin, and Shafiq Joty. 2021. Rethinking coherence modeling: Synthetic vs. downstream tasks. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 3528–3539, Online. Association for Computational Linguistics.

Han Cheol Moon, 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, Hong Kong, China. Association for Computational Linguistics.

Hyeongdon Moon, Yoonseok Yang, Hangyeol Yu, Seunghyun Lee, Myeongho Jeong, Juneyoung Park, Jamin Shin, Minsam Kim, and Seungtaek Choi. 2022. Evaluating the knowledge dependency of questions. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 10512–10526, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

John Morris, Eli Lifland, Jin Yong Yoo, Jake Grigsby, Di Jin, and Yanjun Qi. 2020. TextAttack: A framework for adversarial attacks, data augmentation, and adversarial training in NLP. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 119–126, Online. Association for Computational Linguistics.

Nasrin Mostafazadeh, Nathanael Chambers, Xiaodong He, Devi Parikh, Dhruv Batra, Lucy Vanderwende, Pushmeet Kohli, and James Allen. 2016. A corpus and cloze evaluation for deeper understanding of commonsense stories. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 839–849, San Diego, California. Association for Computational Linguistics.

Jeffrey Pennington, Richard Socher, and Christopher Manning. 2014. GloVe: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1532–1543, Doha, Qatar. Association for Computational Linguistics.

Leila Pishdad, Federico Fancellu, Ran Zhang, and Afsaneh Fazly. 2020. How coherent are neural models of coherence? In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 6126–6138, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Nils Reimers and Iryna Gurevych. 2019. Sentence-BERT: Sentence embeddings using Siamese BERT-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3982–3992, Hong Kong, China. Association for Computational Linguistics.

Victor Sanh, Lysandre Debut, Julien Chaumond, and Thomas Wolf. 2019. Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter. *arXiv preprint arXiv:1910.01108*.

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.

Haoyue Shi, Karen Livescu, and Kevin Gimpel. 2021. Substructure substitution: Structured data augmentation for NLP. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 3494–3508, Online. Association for Computational Linguistics.

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,

Vancouver, Canada. Association for Computational Linguistics.

Teun Adrianus van Dijk. 1977. *Text and Context: Explorations in the Semantics and Pragmatics of Discourse*. Addison-Wesley Longman.

Alex Warstadt, Amanpreet Singh, and Samuel R. Bowman. 2019. Neural network acceptability judgments. *Transactions of the Association for Computational Linguistics*, 7:625–641.

Peng Xu, Hamidreza Saghir, Jin Sung Kang, Teng Long, Avishek Joey Bose, Yanshuai Cao, and Jackie Chi Kit Cheung. 2019. A cross-domain transferable neural coherence model. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 678–687, Florence, Italy. Association for Computational Linguistics.

Jingfeng Yang, Le Zhang, and Diyi Yang. 2022. SUBS: Subtree substitution for compositional semantic parsing. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 169–174, Seattle, United States. Association for Computational Linguistics.

Zhilin Yang, Zihang Dai, Yiming Yang, Jaime G. Carbonell, Ruslan Salakhutdinov, and Quoc V. Le. 2019. Xlnet: Generalized autoregressive pretraining for language understanding. *CoRR*, abs/1906.08237.

Appendix

The appendix presents examples of the informed incoherent set data. Samples from ***M1, M2, M3, M4, M5, M6*** are presented in Tables 8, 9, 10, 11, 12 and 13 respectively. These samples illustrate the systematic application of incoherence strategies such as parse-based substitutions and token manipulation techniques. The appendix aids in understanding the nuanced application of these methods in text analysis.

S1	It was the show's creator Gene Roddenberry who argued in favor of her sudden demise as he felt it was suitable for a security officer.
S2	Roddenberry also argued against killing Armus in retaliation.
S3	Shearer later described the decision, saying Gene felt we couldn't kill the creature, because it is not up to us as human beings to make a moral judgement on any creature that we encounter, because we are not God.
-	
S1	It was the show's creator Gene Roddenberry who argued in favor of her sudden demise as he felt it was suitable for a security officer.
S2	Roddenberry also argued *with Miss Lawson*.
S3	Shearer later described the decision, saying Gene felt we couldn't kill the creature, because it is not *as a kid* to make a moral judgement on any creature that we encounter, because we are not God.
S1	He was confused at first when seeing the cold white snow.
S2	He sniffed and pawed at it at first.
S3	By the end of the day he was jumping around and having fun.
-	
S1	He was confused at first when seeing the cold white snow.
S2	He sniffed and pawed at it at first.
S3	*To an online boggle game* he was jumping around and having fun.
S1	The digging of the ditch coincided with a near famine in Medina.
S2	Women and children were moved to the inner city.
S3	The Medinans harvested all their crops early, so the Confederate armies would have to rely on their own food reserves.
-	
S1	The digging *for a party she is planning* coincided with a near famine in Medina.
S2	Women and children were moved *to the woods*.
S3	The Medinans harvested all their crops early, so the Confederate armies would have to rely on their own food reserves.

Table 8: Examples for ***M1***, **constituency parse tree** based substitutions. The upper half of an example depicts the coherent source and the bottom half depics the perturbed negative window. The pertubations are emphasized.

S1	Gina wanted her brother's room when he left.
S2	Her parents had set it up as a family room.
S3	One day she came home and the family room was moved.
-	
S1	Gina wanted her brother's room when he left.
S2	Her parents had set it up as a family room.
S3	One day she came home and the family *frigate* was *reanimated*.
S1	It begins to feed in the morning, and is more active during the cooler parts of the day.
S2	Loud calls from males indicate the group is ready to move to another tree to feed.
S3	This monkey is mainly a foliovore, and on average, half of the leaves consumed are young leaves.
-	
S1	It begins to feed in the morning, and is more active during the cooler parts of the day.
S2	*plentiful* calls from males indicate the group is ready to *remove* to another *stand* to *evacuate*.
S3	This monkey is mainly a foliovore, and on average, half of the leaves consumed are young leaves.
S1	Capitalizing on the ability of Neural Networks techniques for approximating the solution of PDE's, we incorporate Deep Learning (DL) methods into a DA framework.
S2	More precisely, we exploit the latent structure provided by autoencoders (AEs) to design an Ensemble Transform Kalman Filter with model error (ETKF-Q) in the latent space.
S3	Model dynamics are also propagated within the latent space via a surrogate neural network.
-	
S1	*Rebelling* on the *parent* of *Rats Khalidorans* techniques for approximating the solution of PDE's, we incorporate Deep Learning *(croup)* methods into a DA *arm*.
S2	More precisely, we exploit the latent structure provided by autoencoders (AEs) to design an Ensemble Transform Kalman Filter with model error (ETKF-Q) in the latent space.
S3	Model dynamics are also propagated within the latent space via a surrogate neural network.

Table 9: Examples for ***M2***, **salient Part-of-speech** based substitutions. The upper half of an example depicts the coherent source and the bottom half depics the perturbed negative window. The pertubations are emphasized.

S1	A later meeting at a boat dock in London crushes Gemma's hope that they could be together.
S2	Kartik enlists as a sailor for the HMS Orlando to escape from Gemma and the Rakshana.
S3	He refuses to reveal to Gemma the details of his business with the Rakshana or what he will do beyond being a sailor.
-	
S1	*When the ship is close enough, and the rope high enough above the weed to ensure a safe passage, the narrator rides a breeches buoy to the ship, where he receives a hero's welcome.*
S2	Kartik enlists as a sailor for the HMS Orlando to escape from Gemma and the Rakshana.
S3	He refuses to reveal to Gemma the details of his business with the Rakshana or what he will do beyond being a sailor.
S1	The producers had to contact Spielberg in order to clear the rights for the song so that they could use it in the episode.
S2	Paul Wee was the layout artist for the sequence.
S3	Marge's voice actor, Julie Kavner, praised it for focusing on the animation and not having any dialog in it.
-	
S1	The producers had to contact Spielberg in order to clear the rights for the song so that they could use it in the episode.
S2	Paul Wee was the layout artist for the sequence.
S3	*Presto was directed by veteran Pixar animator Doug Sweetland, in his directorial debut.*
S1	On seeing the captured frames, they shifted all the interior shots to outside.
S2	Filming was completed in 37 days in several locations of Rajasthan.
S3	Since most of the old palaces in Rajasthan have been converted into hotels, the crew stayed at a palace resort called Manwar.
-	
S1	*The tour lasted for four years and travelled to 33 German and Austrian cities.*
S2	Filming was completed in 37 days in several locations of Rajasthan.
S3	Since most of the old palaces in Rajasthan have been converted into hotels, the crew stayed at a palace resort called Manwar.

Table 10: Examples for ***M3***, **semantic similarity** based intruder substitutions. The upper half of an example depicts the coherent source and the bottom half depics the perturbed negative window. The pertubations are emphasized.

S1	Juan was incredibly excited for his first day of middle school.
S2	He had all his supplies and new clothes, and felt prepared.
S3	But the night before, he was so excited he didn't get a wink of sleep.
-	
S1	Juan was incredibly excited for his first day of middle school.
S2	*Brook's first day of school, he mostly sat alone and didn't talk much.*
S3	But the night before, he was so excited he didn't get a wink of sleep.
S1	It was during the time when Premchand first embarked on writing fiction based on contemporary social issues.
S2	Unlike his other works, Nirmala has a darker tone and ending, and its characters are less idealised.
S3	It was translated into English for the first time in 1988.
-	
S1	It was during the time when Premchand first embarked on writing fiction based on contemporary social issues.
S2	Unlike his other works, Nirmala has a darker tone and ending, and its characters are less idealised.
S3	*He said it pushed the boundaries of animation by balancing esoteric ideas with more immediately accessible ones, and that the main difference between the film and other science fiction projects rooted in an apocalypse was its optimism.*
S1	His guide will find him and help him on his quest.
S2	Torak reluctantly leaves his father as the bear comes back to kill him.
S3	Torak heads north and soon encounters an orphaned wolf cub.
-	
S1	His guide will find him and help him on his quest.
S2	Torak reluctantly leaves his father as the bear comes back to kill him.
S3	*They leave and Ivy's father took her out for seafood.*

Table 11: Examples for ***M4***, **salient token overlap** based intruder substitutions. The upper half of an example depicts the coherent source and the bottom half depics the perturbed negative window. The pertubations are emphasized.

S1	When converting lines to electric, the connections with other lines must be considered.
S2	Some electrifications have subsequently been removed because of the through traffic to non-electrified lines.
S3	If through traffic is to have any benefit, time consuming engine switches must occur to make such connections or expensive dual mode engines must be used.
-	
S1	*When lines to electric, the connections converting lines with other must be considered.*
S2	Some electrifications have subsequently been removed because of the through traffic to non-electrified lines.
S3	If through traffic is to have any benefit, time consuming engine switches must occur to make such connections or expensive dual mode engines must be used.
S1	Rene went to the store to buy the meatloaf ingredients.
S2	At home, Rene prepared the meatloaf and baked it.
S3	Rene and her boyfriend had a nice meal together.
-	
S1	Rene went to the store to buy the meatloaf ingredients.
S2	At home, Rene prepared the meatloaf and baked it.
S3	*Rene and meal her boyfriend had a nice together.*

Table 12: Examples for ***M5***, **intra-sentence token permutations**. The upper half of an example depicts the coherent source and the bottom half depics the perturbed negative window. The pertubations are emphasized.

S1	These resonances occur when Neptune's orbital period is a precise fraction of that of the object, such as 1:2, or 3:4.
S2	If, say, an object orbits the Sun once for every two Neptune orbits, it will only complete half an orbit by the time Neptune returns to its original position.
S3	The most heavily populated in the Kuiper with over 200 known objects, is the resonance.
-	
S1	These resonances occur when Neptune's orbital period is a precise fraction of that of the object, such as 1:2, or 3:4.
S2	*If, say, an object orbits the Sun once for two it will only complete half an orbit by the Neptune returns to its position.*
S3	The most heavily populated in the Kuiper with over 200 known objects, is the resonance.
S1	Tommy wanted to get his mom a nice necklace for Christmas.
S2	So he worked a lot during the month of November and December.
S3	He sold a few things from his house for more money.
-	
S1	Tommy wanted to get his mom a nice necklace for Christmas.
S2	*So he a lot during the of November and December.*
S3	He sold a few things from his house for more money.

Table 13: Examples for ***M6***, **context dissipation**. The upper half of an example depicts the coherent source and the bottom half depics the perturbed negative window. The pertubations are emphasized.