Updated Headline Generation: Creating Updated Summaries for Evolving News Stories

Sheena Panthaplackel[1]* **Adrian Benton**[2]† **Mark Dredze**[2,3]
[1] Computer Science, The University of Texas at Austin, TX, USA
[2] Bloomberg, New York, NY USA
[3] Computer Science, Johns Hopkins University, Baltimore, MD USA
spantha@cs.utexas.edu, adbenton@google.com, mdredze@cs.jhu.edu

Abstract

We propose the task of *updated headline generation*, in which a system generates a headline for an updated article, considering both the previous article and headline. The system must identify the novel information in the article update, and modify the existing headline accordingly. We create data for this task using the NewsEdits corpus (Spangher and May, 2021) by automatically identifying contiguous article versions that are likely to require a substantive headline update. We find that models conditioned on the prior headline and body revisions produce headlines judged by humans to be as factual as gold headlines while making fewer unnecessary edits compared to a standard headline generation model. Our experiments establish benchmarks for this new contextual summarization task.

1 Introduction

Automatic text summarization condenses the most important and salient information from a large quantity of text. The task takes many different forms depending on the type of information being summarized, the modality of the information, the type of summary desired and the needs of the end user. Examples include news headline generation (Banko et al., 2000; Zajic et al., 2002; Dorr et al., 2003; Takase et al., 2016; Matsumaru et al., 2020), summarization of social media (Liu et al., 2012; Ding and Jiang, 2015; Kim et al., 2019), and medical documents (Schulze and Neves, 2016; Liang et al., 2019; Adams et al., 2021).

In many settings, users encounter information progressively instead of all at once. For instance, news stories are revised as events unfold (Tannier and Moriceau, 2013), social media streams evolve as people post content (Tarnpradab et al., 2021), and biomedical texts are revised as clinical trial results emerge (uptodate, 2021). In such dynamic settings, existing summaries should be updated as new information becomes available. To address this, we could in principle leverage static summarization systems for generating a summary of the underlying content at any given point in time. However, a more natural approach would be to produce a new summary based on what the reader *already knows* and what content *changed*.

Consider the case of a news article being updated as events unfold (Figure 1). The article first reports that a man is charged with stealing an ice cream van, and the article is later updated when the man admits to the crime. By the time the article is updated, the reader already knows what was stolen, who was charged, and where it happened. At this point, the reader is most interested in what changed, namely the admission of guilt. In the case of news articles, the new headline must both convey critical new information and provide a holistic overview for readers unfamiliar with the story. Updating a summary instead of wholesale replacement falls outside the scope of static summarization systems.

To address these shortcomings, we envision a summarization system that combines an existing summary with information updates. More concretely, following prior work of using headlines as article summaries (Graff et al., 2003), we consider the task of news headline generation. We instead propose *updated headline generation*, which entails *updating* headlines based on changes to the content of the article. In this work, we make the following contributions:

*Work done during an internship at Bloomberg.
†Now at Google Research.

Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics
Volume 1: Long Papers, pages 6438 - 6461
May 22-27, 2022 ©2022 Association for Computational Linguistics

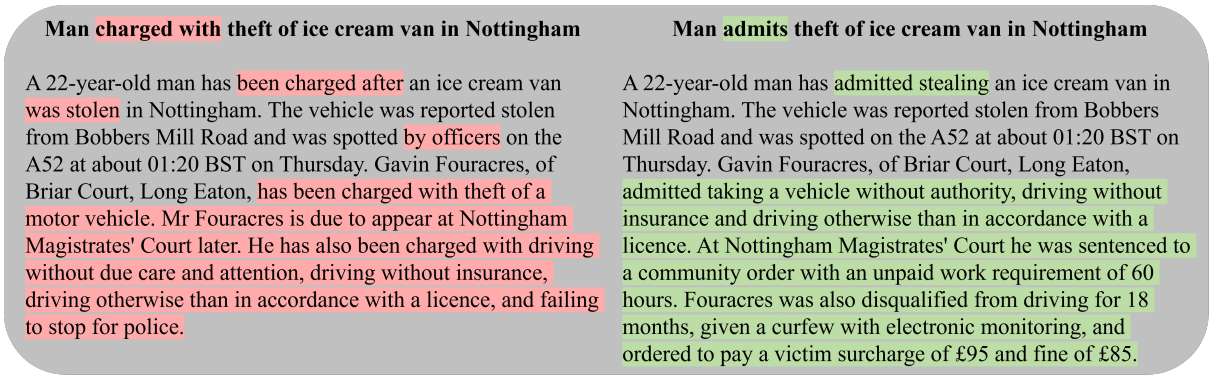

Figure 1: Example of a news story where both the body and headline are revised after publication. The old version of the article is on the left and the revised version is on the right. The body text in red was removed and green text was added as a replacement. Source: https://www.newssniffer.co.uk/articles/1994705/

- Introduce updated headline generation as a model for contextual, dynamic summarization, and support the task with the release of the **H**eadline **R**evision for **E**volving **N**ews dataset (HREN), a subset of the NewsEdits corpus (Spangher and May, 2021) consisting of contiguous article versions.[1]
- Evaluate the contribution of different types of information – previous headline, edits to the article body – to a model that makes updates to an existing news headline.
- Conduct a human evaluation demonstrating that leveraging this additional context leads to headlines which are as factual as standard headline generation models, while applying fewer unnecessary edits.
- Perform an error analysis to determine which types of headline updates are addressed by our model, and what challenges remain.

2 Updated Headline Generation

A news article consists of a body (B) and a headline (H). *Headline generation* (Banko et al., 2000; Zajic et al., 2002; Dorr et al., 2003; Takase et al., 2016; Matsumaru et al., 2020) asks a system to consider B and produce H. We propose *updated headline generation* as a modification of this task. A system receives an existing article (B_1, H_1) and an updated version of the article body (B_2). The goal is to update H_1 to produce a new headline (H_2) that reflects important new information in B_2.

This task introduces several challenges. First, a system must identify the most critical new information in B_2. Changes to the article can be small or very significant, and it must determine which of these changes, if any, should be reflected in the headline. Second, it needs to consider *how* to modify H_1. Oftentimes a revision to an article will preserve most of the structure of H_1, even if a completely rewritten headline might convey the same information. New information should be reflected in an updated headline with minimal edits, for the sake of continuity and minimizing cognitive load on a reader who is following an evolving story. Third, there are different types of updated stories that each require a different style of headline update. Stories can be updated as the underlying event progresses (e.g., criminal investigations, natural disasters, voting on legislation or appointments, live events), new or corrected information becomes available (e.g., number of people injured following an accident), or public figures react to the event (e.g., political figure commenting on a situation). See Table 1 for examples.

3 Dataset

The NewsEdits (Spangher and May, 2021) corpus contains articles with revision histories derived from 22 wires: 5 from *News Sniffer*[2] and the remainder from Twitter accounts powered by *Diffengine*.[3] It consists of over one million articles with 4.6 million revisions. In this work, we focus on the 5 English language wires from *News Sniffer* (Washington Post, NY Times, Independent, BBC, Guardian), as we found them to have cleaner revision histories.

From the revision history of a given article, we extract body-headline pairs by examining consecutive versions, (B_k, H_k), (B_{k+1}, H_{k+1}), resulting in examples of the form {(B_1, H_1), (B_2, H_2)}. We

[1] Available at: https://github.com/panthap2/updated-headline-generation

[2] https://www.newssniffer.co.uk/

[3] https://github.com/DocNow/diffengine

B_1	H_1	B_2	H_2
Nearly a million people in southern Vietnam face **evacuation from the path of a deadly tropical storm**...	Tembin: **Vietnam braces for killer storm**	A tropical storm that **was threatening southern Vietnam has weakened and is expected to dissipate**...	Tembin: **Storm weakens as it nears southern Vietnam**
A no-confidence motion in Wales' health minister over Cwm Taf's maternity service failings **has been debated by AMs and will be voted on later**....	Cwm Taf: Health minister **facing** no-confidence vote	Wales' health minister **has survived** a Plaid Cymru no-confidence motion in him after severe failings were uncovered at Cwm Taf's maternity services...	Cwm Taf: Health minister **survives** no-confidence vote
US astronauts Doug Hurley and Bob Behnken **will dock** to the International Space Station (ISS) **in the next hour**...	SpaceX Nasa Mission: Astronaut capsule **closes in on** space station	US astronauts Doug Hurley and Bob Behnken **have docked** with the International Space Station (ISS)...	SpaceX Nasa Mission: Astronaut capsule **docks with** space station
At least **19** people were injured in a crash involving a charter bus and a tractor-trailer on a Virginia interstate on Sunday morning, the authorities said...	At Least **19** Hurt in Tractor-Trailer and Bus Crash on I-64 in Virginia	At least **24** people were injured in a crash involving a charter bus and a tractor-trailer on a Virginia interstate on Sunday morning, the authorities said...	At Least **24** Hurt in Tractor-Trailer and Bus Crash on I-64 in Virginia
...special counsel Robert Mueller, the man charged with investigating Russian interference in the US election and possible collusion with Trump's campaign, **with one friend of the president floating the possibility he could fire Mueller.**	Trump **may sack** special counsel in Russia inquiry, **says friend**	**Rod Rosenstein**, the deputy attorney general, has **hit back following speculation** that Donald Trump was considering firing the special counsel Robert Mueller, assuring senators he was aware of **"no secret plan"** to oust the former FBI director...	**Rod Rosenstein: 'no secret plan' to fire** special counsel in Trump-Russia inquiry

Table 1: Examples of evolving news stories, with important changes between B_1 and B_2, and H_1 and H_2 in **bold**.

exclude cases without a change in the body, and group examples into two different classes: *positive*– examples where the headline is updated (i.e., $H_1 \neq H_2$) – and *negative*– the headline remains unchanged (i.e., $H_1 = H_2$). We observed that the headline change associated with a particular body change sometimes occurred in the subsequent revision (not contemporaneous). So, we also include positive examples which have the following property across three consecutive revisions: only the body is changed between the first and second versions and only the headline is changed between the second and third versions, i.e., (B_1, H_1) $\rightarrow$ (B_2, H_1) $\rightarrow$ (B_2, H_2). We do not include (B_1, H_1) $\rightarrow$ (B_2, H_1) as a *negative* example for such cases.

To avoid spurious *positive* examples, we tried removing versions that were incorrectly paired together,[4] or where the headline change was trivial.[5] This process produced a dataset of 144,218 positive and 794,372 negative examples. Even after filtering by heuristic, we found that many of the remaining headline changes still do not reflect a substantive update to the article. These include purely stylistic changes, embellishments, and rephrasings.

	Train	**Valid**	**Test**
# Examples	57,285	5,769	6,189
# Tokens			
H_1	9.0	10.9	10.9
H_2	9.2	11.2	11.2
B_1	479.6	576.5	699.3
B_2	574.4	716.3	846.4
B_{edits}	807.0	965.3	1129.5
B_{edits} (change only)	458.2	530.5	593.2

Table 2: Avg # of examples and tokens/document in HREN.

To filter such cases, we develop a classifier which is trained to determine whether H_1 needs to be updated based on the changes between B_1 and B_2. The classifier achieves 51.9 F1, indicating that this is a challenging problem; the training and evaluation data are silver-labeled, and noisy. We filter the remaining *positive* examples with this classifier. Empirically, we find that training on this filtered subset leads to improved performance. We provide a complete description of the classifier and attendant experiments in Appendix A.

[4] B_1 and B_2 are sometimes completely unrelated, likely due to an error in the News Sniffer collection. We removed examples in which B_2 was published more than a week after B_1, and we exclude articles that yield more than 8 version pairs (95th percentile).

[5] Trivial headline changes included modifications limited to spacing and punctuation, as well as simple rephrasing (i.e., changes to stopwords or the surface form of a lemma).

3.1 The HREN Dataset

After data cleaning and filtering, we obtain the **H**eadline **R**evision for **E**volving **N**ews dataset (HREN), which contains 69,243 examples with meaningful headline edits. Descriptive statistics for each fold are listed in Table 2. Average number of tokens per document are broken by source text type, with B_{edits} and B_{edits} (change only) described in Section 4.2.

We partition the data into 80/10/10 training, validation, and test splits. While constructing the data, we took care to ensure that the underlying articles from which examples are drawn are disjoint for partitions, and that the timestamps corresponding to examples in the training set strictly precede those in the validation set, which in turn precede those in the test set.[6] This ensures that we train on strictly historical data. Our main experiments use HREN, though we include negative examples and filtered positive examples in some later experiments.

4 Sources of Information

We study the importance of several types of information – in the form of baselines and inputs to models – for updated headline generation.

4.1 Rule-based Baselines

COPY H_1: Updated headlines usually copy parts of the original headline and the overall structure. For instance, in Figure 1, 8 of the 9 tokens in the updated headline come from the original one. So, we consider copying H_1 as the prediction.

LEAD-1: Newsroom style guides dictate that the most significant information should appear first (Siegal and Connolly, 1999). Consequently, the lead sentence typically includes information that is mentioned in the headline, as shown in Figure 1. This baseline uses the lead sentence of B_2 as the prediction for H_2.

SUBSTITUTION: Many headlines can be correctly updated by a simple token replacement, reflecting an analogous replacement in the body. Table 1: H_1 is "At least **19** Hurt in Tractor-Trailer and Bus crash on I-64 in Virginia" and a sentence in B_1 "At least **19** people..." is updated to "At least **24** people" in B_2, prompting a similar change in the headline H_2. So, if a single token (t_1=**19**) appearing in both H_1 and B_1 is replaced with a new token (t_2=**24**) in B_2, we form H_2 by substituting t_1 with t_2 in H_1. We only consider single-token replacements and copy H_1 if a substitution cannot be made. Note that this is a high precision baseline, with 10.8% of headlines able to be updated by this heuristic.

[6] Similar time-based partitioning was done for the classifier. See Appendix B for date cutoffs for each fold.

4.2 Context Representations

We study various configurations for representing the input context for training the models.

H_1: Many headline updates follow a natural progression of events (e.g., "Lori Loughlin Expected to Plead Guilty via Zoom in College Admissions Case" → "Lori Loughlin Pleads Guilty via Zoom in College Admissions Case"). In these cases, knowing the old headline may be sufficient to predict the subsequent headline. Therefore, we consider providing only H_1 to a statistically trained model.

B_2: This is the standard headline generation setting in which a model must predict H_2 given B_2.

H_1 + B_2: We provide both H_1 and B_2. Faithfulness to the article body is paramount for automatic headline generation (Matsumaru et al., 2020), and leveraging the original headline removes some of the burden of generating a headline from scratch.

H_1 + B_2 + B_1: We provide all available context to the model, so that the model can compare story versions and consider the old headline during decoding.

H_1 + B_{edits}: Asking the model to compare two full articles may be unrealistic. Instead, we provide the sequence of edits between B_1 and B_2:

<KEEP> A 22-year old man has <KEEP_END>
<REPLACE_OLD> been charged after
<REPLACE_NEW> admitted stealing
<REPLACE_END>
<KEEP> an ice cream...

This sequence consists of edit actions: *insert*, *delete*, *replace*, and *keep*, and are represented in the format proposed by Panthaplackel et al. (2020). We study whether providing explicit body edits helps a model learn to apply analogous headline edits.

H_1 + B_{edits} (change only): Rather than feeding in the full edit sequence, we discard *keep* spans. While this removes information about *where* the edits are made, it significantly reduces the amount of context a model must reason about (Table 2).

5 Models

We evaluate two encoder-decoder models that utilize each of the representations described in Section 4.2. Note that we first preprocess all representations using the Penn Treebank tokenizer[7] to tokenize and split text into sentences and words, prior to model-specific preprocessing.

Pointer Networks consist of separate LSTM encoders for body and headline text, and these are concatenated to form the initial states for an LSTM decoder, equipped with attention (Vinyals et al., 2015; Wang et al., 2016). The hidden states are concatenated for both attention and copy mechanisms. We posit that this model might be effective at headline updating, as this task benefits from copying tokens from the input context (especially H_1). We initialize embeddings for the model with GloVe (Pennington et al., 2014).

BART (Lewis et al., 2020) is a pretrained transformer network considered state-of-the-art for summarization. Because we focus on the news domain, we consider a version of BART already fine-tuned for summarization on news articles from CNN-Daily Mail (Hermann et al., 2015).[8] We further fine-tune on our data, by concatenating inputs into a single sequence, separated by special tokens (e.g., <OLD_HEADLINE>, <NEW_BODY>).

We evaluate all context representations with both of these architectures, with the exception of H_1 + B_2 + B_1 for BART, due to limitations in fitting the entire input context within BART's 1024 token limit. We use beam search with a beam size of 20 to decode for all models along with bigram blocking (Paulus et al., 2018).[9] These decoding hyperparameters were found to work well across models during preliminary experimentation based on an unweighted average across automated metrics.

6 Experiments

We evaluate with common text-generation metrics: METEOR (Banerjee and Lavie, 2005), ROUGE-L (Lin and Och, 2004), and BLEU-4 (Papineni et al., 2002). Given the editing nature of our task, we also use two edit-specific metrics: GLEU (Napoles et al., 2015) and SARI (Xu et al., 2016). SARI measures the average n-gram F1 scores corresponding to edit operations (add, delete, and keep). GLEU closely follows BLEU except that it places more importance on n-grams which have been correctly changed. We compute statistical significance at the $p < 0.05$ level using bootstrap tests (Berg-Kirkpatrick et al., 2012).

Rule-Based Baselines: Our results (Table 3) show that rule-based baselines achieve relatively high performance, even beating the headline generation setting (B_2) for the pointer network and BART in some cases. Due to the high lexical overlap between H_1 and H_2, the COPY H_1 baseline can perform well on automated metrics, specifically the three text-generation metrics. The SUBSTITUTION baseline performs slightly better than simply copying H_1 by making simple substitutions in 10.8% of examples, demonstrating improvements in the two edit-based metrics. The LEAD-1 baseline performs lower than the other baselines on most metrics due to the discrepancy between the structure and style of the lead sentence and headlines, with the average lead sentence length being 36.7 tokens. However, the SARI score is substantially higher.[10]

Using H_1: For both the pointer network and BART, providing only H_1 results in lower performance than COPY H_1 for most metrics, except for SARI, which is designed to evaluate edits. Higher SARI suggests that these models are able to make the necessary edits in some cases by guessing the natural progression of events, without the news body, such as forecasting the order of events following a police investigation (e.g., suspect is arrested, charged, and then appeared in court). However, the SARI score is still much lower than if only B_2 is provided, as in standard headline generation. This highlights the importance of the latest version of the article body in updated headline generation. Nonetheless, by comparing performance of B_2 and H_1 + B_2 across both architectures, we see the extent to which H_1 can guide headline generation models in selecting important content and determining structure for the output. This demonstrates the inadequacy of framing this as a static headline generation task. The improvements on edit metrics are more limited because a model which has access

[7] https://nlp.stanford.edu/nlp/javadoc/javanlp/edu/stanford/nlp/process/PTBTokenizer.html

[8] https://huggingface.co/facebook/bart-large-cnn

[9] We chose bigram instead of the more typical trigram blocking as headlines tend to be short.

[10] SARI is calculated as the average of N-gram F1 scores of add, delete, and keep edit operations. Because the lead sentence will not contain many n-grams in H_1 that should be deleted, and will contain some n-grams that should be inserted into H_2, the SARI score is high for this baseline.

		METEOR	ROUGE-L	BLEU-4	GLEU	SARI
Rule-Based	COPY H_1	29.9	49.9	35.1	19.2	14.6
	LEAD-1	20.8	21.7	7.3	6.0	33.0
	SUBSTITUTION	31.1	50.0	35.5	20.8	18.8
Pointer Network	H_1	26.1	45.9	30.9	18.8	26.9
	B_2	14.2	26.9	14.8	14.6	30.6§
	H_1 + B_2	28.8	48.6	33.4	20.3	26.5
	H_1 + B_2 + B_1	29.0	48.7	33.7	19.6	22.8
	H_1 + B_{edits}	29.6	49.3	34.1	21.2‡	29.2
	H_1 + B_{edits} (change only)	29.0	49.0	33.5	21.0‡	30.5§
BART	H_1	29.4	48.8	34.0	19.6	21.3
	B_2	21.4	35.2	20.2	17.7	35.6
	H_1 + B_2	32.6	51.5	35.2	25.2	**40.1**†
	H_1 + B_{edits}	32.5	50.3	34.7	23.2	34.6
	H_1 + B_{edits} (change only)	**34.0**	**52.4**	**36.5**	**26.0**	39.7†

Table 3: Test performance of headline updating models on HREN. Results for the best model in each of the three model classes are underlined. For each category, differences between underlined scores and all other scores which are NOT statistically significant ($p < 0.05$) are indicated with matching symbols. The best model for each metric is **bolded**. Bolded scores are statistically significantly higher than scores for all rule-based and pointer network models across all metrics.

to H_1 will learn to copy many parts of this input, and consequently will make fewer edits.

Using Body Edits: To investigate whether providing body edits can further improve performance by helping a model learn to correlate them with H_1 and apply analogous updates, we consider different ways of incorporating B_1. First, in the pointer network, we evaluate performance when just feeding it in as another input (H_1 + B_2 + B_1). We observe no improvement in performance, suggesting that the model fails to implicitly learn the edits. Next, we consider collapsing B_1 and B_2 into a sequence of edits (H_1 + B_{edits}), with which we see a slight improvement in performance over H_1 + B_2 for the pointer network but a reduction in performance for BART. We believe this is because BART struggles to model longer input sequences. When we reduce the context length and provide only the changes in the edit sequence (H_1 + B_{edits} (change only)), we see an improvement in BART. Note that the performance of H_1 + B_{edits} (change only) is lower for the pointer network across most metrics. This may be due to a lack of pretraining, whereas BART is already equipped with a strong language model.

Pointer Network vs. BART: While both model classes perform well, BART models tend to perform better overall, demonstrating the value of BART's larger transformer-based architecture and pretraining. Nonetheless, the benefits of using H_1 and body edits generalize across both architectures. We expect that the performance of more recent summarization models such as PEGASUS (Zhang et al., 2020) or SimCLS (Liu and Liu, 2021) will exhibit a similar trend as BART, but we welcome evaluation of other large pretrained summarization models on HREN.

	Fact†	Focs†	MnEd†	Hdln	Grm*
Cpy H_1	4.63	4.26	**4.96**$^{\sharp\natural}$	4.97	**5.00**
B_2	4.88$^{\flat}$	4.67$^{\flat}$	1.86	4.96	4.97
H_1+B_2	4.90$^{\flat}$	**4.71**$^{\flat}$	3.15$^{\natural\sharp}$	**4.98**	4.95
H_1+B_{ed} (ch only)	4.81$^{\flat}$	4.64$^{\flat}$	3.35$^{\natural\sharp}$	4.96	4.95
Gold	**4.92**$^{\flat}$	**4.71**$^{\flat}$	2.30$^{\natural}$	4.96	4.98

Table 4: Human evaluation results. Differences that are statistically significant by Tukey HSD at the $p < 0.05$ level are indicated by superscripts. Superscripts indicate that the model is significantly better than COPY H_1 $^{\flat}$, Gold $^{\sharp}$, or B_2 $^{\natural}$. Best average score for each item is in **bold**. ANOVA statistical significance level is indicated on the column header ($* : p < 0.05$, $\dagger : p < 10^{-10}$).

6.1 Human Evaluation

Design We conduct a human evaluation of the (more performant) BART models with the following configurations: B_2, H_1 + B_2, H_1 + B_{edits} (change only). As points of reference, we also evaluate the gold headline (H_2) and the output of the COPY H_1 baseline.

Annotators were presented with a visual *diff* between B_1 and B_2 along with H_1, and were asked to judge a candidate updated headline according to five dimensions on a Likert scale – whether the updated headline was **factual**, **grammatical**, appears to be written in **headlinese**, **focus**es on important changes/information in the updated body (similar to the relevance criterion commonly used to evaluate natural language generation models (Sai et al., 2020)), and makes only **minimal edits** to the original headline. We introduce the last dimension since we frame our task as an *editing* task. The underlying idea behind editing is that change should only be made to be parts that warrant it; all other

parts that do not need to be changed should be preserved, which is consistent with how humans edit text (Panthaplackel et al., 2020). Additionally, this is consistent with the task motivation, in which we expect a reader to interpret the important changes in a minimally edited headline with less cognitive effort. We sampled 200 test examples, 143 from HREN and 57 from the unfiltered sample,[11] resulting in 806 unique annotation tasks.[12] Each task was independently annotated by three paid annotators who were trained on this task – native English speakers, two of whom were journalism majors. See Appendices C and D for more details on the annotation procedure.

Results We present average annotator ratings for each dimension in Table 4. Following the human evaluation analyses in Reiter and Belz (2009) and Wiseman et al. (2021), we compute statistical significance using multi-way ANOVA tests, followed by Tukey's post hoc HSD tests for pairwise statistical significance (at the $p < 0.05$ level).

For **headlinese** and **grammatical**, we find no significant difference between the approaches; all achieve relatively high scores. With respect to **factual** and **focus**, all approaches perform similarly except for COPY H_1 which significantly underperforms the others, by inaccurately reflecting the state of matters after the story is updated and failing to highlight important changes in B_2. On the other hand, COPY H_1 achieves the best performance on **minimal edits** by definition (i.e., H_1 has minimal edits with itself). As expected, without access to H_1, the headline generation model (B_2) achieves the lowest performance on this dimension. Overall, we find that the two BART models which also include H_1 as context performed better, even beating gold headlines on this dimension. This is unsurprising as gold headlines often undergo stylistic rewrites, in addition to reflecting changes to the facts of evolving news stories. For example, *Byron Burger Menu 'Reassured' Allergy Death Owen Carey* is rewritten as *Byron Burger Death: Owen Carey's Family Demand Law Change* – the form of the headline changes in addition to the release of new information. Although H_1 + B_{edits} (change only) performs slightly better than H_1 + B_2 on automated metrics in Table 3, we find that they perform similarly on the five dimensions.

In summary, incorporating H_1 leads to predictions which make fewer unnecessary edits to the original headline, while simultaneously maintaining performance with respect to factuality, focus, headlinese, and grammaticality of headline generation models (on par with gold headlines).

Input	Prediction
H_1	man remanded over theft of ice cream van in nottingham
B_2	gavin fouracres admits stealing ice cream van in nottingham
H_1 + B_2	man, 22, admits theft of ice cream van in nottingham
H_1 + B_{edits}	man pleads guilty to theft of ice cream van in nottingham
H_1 + B_{edits} (change only)	man admits theft of ice cream van in nottingham
Gold	man admits theft of ice cream van in nottingham

Table 5: Predictions for BART under different input representations, for the example in Figure 1.

7 Discussion

Case Study Table 5 presents BART predictions for the example in Figure 1 under different context representations.[13] Given only H_1, the model predicts an updated headline by speculating about what might follow a person being charged with a crime. Using only B_2, the model generates a headline which reflects that a person has admitted to the crime, but it deviates from the form of the original headline by inserting the name of the person and altering terminology. These aspects of the story have not changed, and should not be changed in the headline. With H_1 + B_2, the prediction captures the major change in the article and better retains the form of H_1, but it still makes an unnecessary change by inserting the person's age into the headline. Given H_1 + B_{edits}, the model learns to only edit the part which is relevant to the body changes, but the terminology used to perform the edit (i.e., *pleads guilty*) varies from the article (*admits* to the crime.) In contrast, H_1 + B_{edits} (change only) is able to simultaneously perform minimal edits and correlate edits between the article and headline.

Performance by Edit Level Headlines require more extensive edits when there are more substantial changes to the article. We perform a fine-grained analysis to better understand how various context representations fare for these different types of examples. We group examples based on

[11] Results on the unfiltered examples are in Appendix A.3.

[12] Models with identical predictions were joined as the same task, and the annotator scores for this task were assigned to all generating models.

[13] Additional examples are provided in Appendix F.

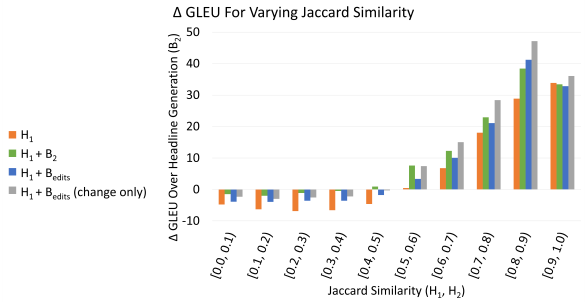

Figure 2: Absolute difference in GLEU between various BART models and standard headline generation (B_2) for each Jaccard headline similarity bucket.

the Jaccard similarity between H_1 and the gold H_2; low similarity means significant edits, high similarity means minimal edits. For the BART-based model we calculate the GLEU score for each bucket because we find that it is better suited for simultaneously evaluating whether appropriate edits were made along with generation quality.

Figure 2 shows the change in performance attributed to each of the context representations relative to headline generation (B_2). For low similarity values, none of the specialized context representations outperform standard headline generation. This suggests that when more substantial edits are needed, starting from scratch may be best. As the similarity increases, models which utilize H_1 perform substantially better. For moderate similarity, having B_2 instead of body edits performs marginally better, but this changes as the similarity score increases, with B_{edits} (change only) leading to drastic improvements.

Analyzing Attention To better understand how models explicitly make use of the old headline, we analyze how the H_1 + B_{edits} (change only) BART model's decoder attends to H_1. For this, we follow the methodology of Vig and Belinkov (2019). Namely, we label each context token by which span it belongs to: H_1, one of the edit spans, or a span delimiter token (*Other*). We compute the average attention paid to each context token class by the BART decoder across all examples and layers. We find that even though only 1.5% of the context tokens are from H_1, they attract over 17% of the BART decoder's attention.

Interestingly, the attention paid to added content and headline tokens increases in later layers at the expense of Other tokens (Figure 3). We posit that this is because the initial layers need to attend to special tag tokens in order to understand which type of span each enclosed token belongs to. This may also arise from the fact that initially the decoder attends to all tokens relatively uniformly (49.9% of tokens are *Other* on average). However, even in the initial layers, the tokens in H_1 are attended to more than would be expected by a uniform attention distribution, likely because H_1 text always appears near the start of the context. Because of this, locating the H_1 tokens is less dependent on identifying enclosing tags – absolute position also helps.

We also find that the decoder attends to tokens in H_1 more often than would be expected under a uniform attention model, *until* it needs to refer to a new piece of information that was added to the article body. Figure 4 displays the relative attention paid to each token type for a decoded headline exemplifying this phenomenon. See Appendix E for additional detail on the decoder attention analysis.

Error Cases Finally, we inspected cases where annotators assigned very low or high scores. We observe with B_2 alone, the headline generation model makes factual errors by mixing up important details when two similar types of entities are discussed in the article (e.g., mixing up the victim and suspect of a crime, mixing up locations and dates).

Additionally, it makes factual errors by omitting something important, which drastically changes the meaning (e.g., missing a letter in the acronym for an organization). On the other hand, because H_1 often includes important background that can be directly copied, we find fewer such factual errors caused by omission for the H_1 + B_2 and H_1 + B_{edits} (change only) models. Having H_1 also helps in maintaining important details (e.g., event location) and specifying the level of detail that is needed.

In general, H_1 is most useful when there is high lexical overlap with the lead sentences of B_2. If the content is significantly different (e.g., the focus of the article changes), it becomes less useful and can even hurt performance in some cases, since H_2 is likely very different from H_1. Body edits are most useful when there are few edits and these edits can be easily grounded in H_1. For H_1 + B_{edits} (change only), we also noticed errors where the model incorrectly correlates body edits with H_1, resulting in it erroneously inserting body tokens that are edited into the headline.

8 Related Work

Summarization: Summarization is a widely studied topic in the NLP community, with multiple

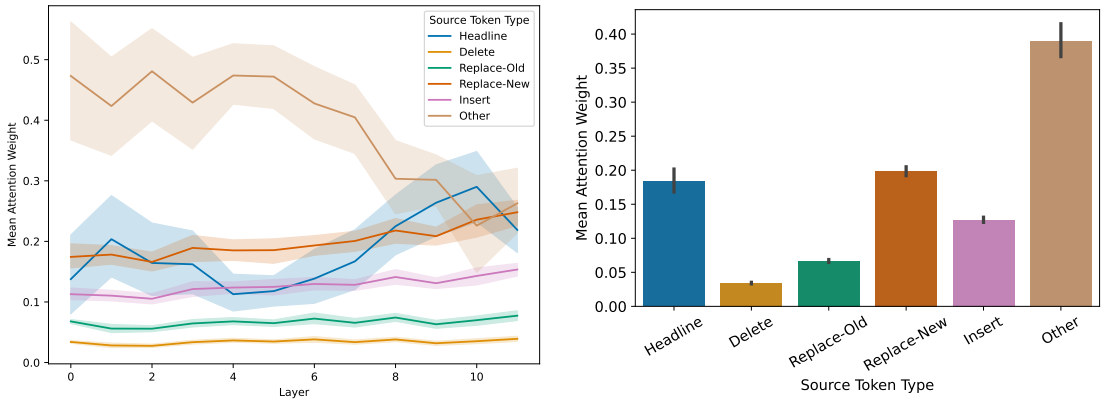

Figure 3: Mean and 95% confidence interval of attention for each token type per decoder layer (left) and averaged across all layers (right).

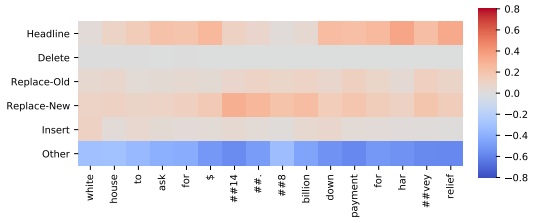

Figure 4: Difference between average attention placed on each span type during decoding of H_2 and that expected under a model that attended uniformly to all tokens. X-axis: words in the decoded headline; Y-axis: types of context tokens. Red cells indicate that a token type is being attended to more than would be expected under uniform attention, whereas blue cells indicate the opposite. H_1: *White House to Ask for $12 Billion Down Payment for Harvey Relief*, Source: https://www.newssniffer.co.uk/articles/1447256.

subtopics relevant to our task. For instance, *multi-document summarization* (Barzilay and McKeown, 2005) pertains to generating a unified summary by synthesizing non-redundant content from multiple related documents. In our setting, we consider multiple documents (i.e., the old and new versions of an article) as well, but we also have an existing summary, and our task requires reasoning about how the non-redundant content from the newer version of the article affects this existing summary. With *update summarization* (Dang et al., 2008), there is an older set of documents as well as a newer set of documents, and the goal is to generate a summary which captures only added and changed information. In contrast, our task aims to incorporate these changes into an already existing holistic summary.

Natural language edits: Our work focuses on learning from edits in news articles to apply update and existing headline. Prior work studies the nature of edits in various texts including news (Faigley and Witte, 1981; Tamori et al., 2017) and Wikipedia (Yang et al., 2017; Faruqui et al., 2018). There has also been extensive work on generating edits for tasks such as grammatical error correction (Bryant et al., 2019), sentence simplification (Zhu et al., 2010), style transfer (Fu et al., 2018), fact-based sentence editing (Shah et al., 2020; Iso et al., 2020), text improvement (Tanaka et al., 2009), and comment updating based on source code changes (Panthaplackel et al., 2020).

9 Conclusion

In this work, we show that headline generation models can benefit from access to the past state of the article. Our proposed model, $H_1 + B_{edits}$ (change only), can generate headline predictions that are statistically tied with gold headlines in terms of factuality, while making fewer unnecessary edits. By releasing the HREN dataset, we hope to encourage the community to produce higher quality tools for aiding journalists, as well as encourage research in NLP over dynamic texts.

Acknowledgements

Sheena Panthaplackel receives support from Bloomberg's Data Science Ph.D. Fellowship Program. We would like to thank Alex Spangher for sharing an early version of the NewsEdits corpus. We would also like to thank the Bloomberg AI group and Lina Vourgidou for early feedback on this project, and illuminating conversations on the application of machine learning and natural language processing to the newsroom. Finally, we thank the reviewers for their constructive feedback.

References

Griffin Adams, Emily Alsentzer, Mert Ketenci, Jason Zucker, and Noémie Elhadad. 2021. What's in a summary? laying the groundwork for advances in hospital-course summarization. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4794–4811.

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.

Michele Banko, Vibhu O Mittal, and Michael J Witbrock. 2000. Headline generation based on statistical translation. In *Proceedings of the 38th Annual Meeting of the Association for Computational Linguistics*, pages 318–325.

Regina Barzilay and Kathleen R McKeown. 2005. Sentence fusion for multidocument news summarization. *Computational Linguistics*, 31(3):297–328.

Taylor Berg-Kirkpatrick, David Burkett, and Dan Klein. 2012. An empirical investigation of statistical significance in NLP. In *Proceedings of the 2012 Conference on Empirical Methods in Natural Language Processing*, pages 995–1005.

Christopher Bryant, Mariano Felice, Øistein E Andersen, and Ted Briscoe. 2019. The BEA-2019 shared task on grammatical error correction. In *Proceedings of the Fourteenth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 52–75.

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*, pages 7282–7296.

Daniel Dahlmeier, Hwee Tou Ng, and Siew Mei Wu. 2013. Building a large annotated corpus of learner English: The NUS corpus of learner English. In *Proceedings of the Eighth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 22–31.

Hoa Trang Dang, Karolina Owczarzak, et al. 2008. Overview of the TAC 2008 update summarization task. In *Text Analysis Conference*.

Johannes Daxenberger and Iryna Gurevych. 2013. Automatically classifying edit categories in Wikipedia revisions. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 578–589.

Ying Ding and Jing Jiang. 2015. Towards opinion summarization from online forums. In *Proceedings of the International Conference Recent Advances in Natural Language Processing*, pages 138–146.

Bonnie Dorr, David Zajic, and Richard Schwartz. 2003. Hedge Trimmer: a parse-and-trim approach to headline generation. In *Proceedings of the 2003 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies on Text summarization workshop*, volume 5, pages 1–8.

Lester Faigley and Stephen Witte. 1981. Analyzing revision. *College composition and communication*, 32(4):400–414.

Manaal Faruqui, Ellie Pavlick, Ian Tenney, and Dipanjan Das. 2018. WikiAtomicEdits: A multilingual corpus of Wikipedia edits for modeling language and discourse. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 305–315.

Zhenxin Fu, Xiaoye Tan, Nanyun Peng, Dongyan Zhao, and Rui Yan. 2018. Style transfer in text: Exploration and evaluation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, pages 663–670.

David Graff, Junbo Kong, Ke Chen, and Kazuaki Maeda. 2003. English Gigaword. *Linguistic Data Consortium, Philadelphia*, 4(1):34.

Karl Moritz Hermann, Tomás Kociský, Edward Grefenstette, Lasse Espeholt, Will Kay, Mustafa Suleyman, and Phil Blunsom. 2015. Teaching machines to read and comprehend. In *Advances in Neural Information Processing Systems*, pages 1693–1701.

Hayate Iso, Chao Qiao, and Hang Li. 2020. Fact-based text editing. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 171–182.

Byeongchang Kim, Hyunwoo Kim, and Gunhee Kim. 2019. Abstractive summarization of Reddit posts with multi-level memory networks. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2519–2531.

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.

Jennifer Liang, Ching-Huei Tsou, and Ananya Poddar. 2019. A novel system for extractive clinical note summarization using EHR data. In *Proceedings of the 2nd Clinical Natural Language Processing Workshop*, pages 46–54.

Chin-Yew Lin and FJ Och. 2004. Looking for a few good metrics: ROUGE and its evaluation. In *NII Testbeds and Community for Information Access Research Workshop*.

Xiaohua Liu, Yitong Li, Furu Wei, and Ming Zhou. 2012. Graph-based multi-tweet summarization using social signals. In *Proceedings of the 24th International Conference on Computational Linguistics*, pages 1699–1714.

Yixin Liu and Pengfei Liu. 2021. SimCLS: A simple framework for contrastive learning of abstractive summarization. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 1065–1072.

I Mårdh. 1980. Headlinese: On the grammar of English front page headlines (vol. 58). *Liberläromedel/Gleerup*.

Kazuki Matsumaru, Sho Takase, and Naoaki Okazaki. 2020. Improving truthfulness of headline generation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1335–1346.

Courtney Napoles, Keisuke Sakaguchi, Matt Post, and Joel Tetreault. 2015. Ground truth for grammatical error correction metrics. In *Annual Meeting of the Association for Computational Linguistics and the International Joint Conference on Natural Language Processing*, pages 588–593.

Sheena Panthaplackel, Pengyu Nie, Milos Gligoric, Junyi Jessy Li, and Raymond Mooney. 2020. Learning to update natural language comments based on code changes. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1853–1868.

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.

Romain Paulus, Caiming Xiong, and Richard Socher. 2018. A deep reinforced model for abstractive summarization. In *International Conference on Learning Representations*.

Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014. GloVe: Global vectors for word representation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*, pages 1532–1543.

Ehud Reiter and Anja Belz. 2009. An investigation into the validity of some metrics for automatically evaluating natural language generation systems. *Computational Linguistics*, 35(4):529–558.

Ananya B Sai, Akash Kumar Mohankumar, and Mitesh M Khapra. 2020. A survey of evaluation metrics used for NLG systems. *arXiv preprint arXiv:2008.12009*.

Stephanie Schoch, Diyi Yang, and Yangfeng Ji. 2020. "This is a problem, don't you agree?" framing and bias in human evaluation for natural language generation. In *Proceedings of the 1st Workshop on Evaluating NLG Evaluation*, pages 10–16.

Frederik Schulze and Mariana Neves. 2016. Entity-supported summarization of biomedical abstracts. In *Proceedings of the Fifth Workshop on Building and Evaluating Resources for Biomedical Text Mining*, pages 40–49.

Darsh Shah, Tal Schuster, and Regina Barzilay. 2020. Automatic fact-guided sentence modification. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 8791–8798.

Allan M Siegal and William G Connolly. 1999. *The New York Times manual of style and usage*. Three Rivers Press (CA).

Alexander Spangher and Jonathan May. 2021. NewsEdits: A dataset of revision histories for news articles (technical report: data processing). *arXiv preprint arXiv:2104.09647*.

Heinrich Straumann. 1935. *Newspaper headlines: A study in linguistic method*. London, Allen.

Sho Takase, Jun Suzuki, Naoaki Okazaki, Tsutomu Hirao, and Masaaki Nagata. 2016. Neural headline generation on abstract meaning representation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1054–1059.

Hideaki Tamori, Yuta Hitomi, Naoaki Okazaki, and Kentaro Inui. 2017. Analyzing the revision logs of a Japanese newspaper for article quality assessment. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing Workshop: Natural Language Processing meets Journalism*, pages 46–50.

Hideki Tanaka, Akinori Kinoshita, Takeshi Kobayakawa, Tadashi Kumano, and Naoto Katoh. 2009. Syntax-driven sentence revision for broadcast news summarization. In *Proceedings of the 2009 Workshop on Language Generation and Summarisation*, pages 39–47.

Xavier Tannier and Véronique Moriceau. 2013. Building event threads out of multiple news articles. In *Proceedings of the 2013 Conference on Empirical Methods in Natural Language Processing*, pages 958–967.

Sansiri Tarnpradab, Fereshteh Jafariakinabad, and Kien A Hua. 2021. Improving online forums summarization via hierarchical unified deep neural network. *arXiv preprint arXiv:2103.13587*.

uptodate. 2021. UpToDate. https://www.uptodate.com/contents/search. Accessed: 2021-11-03.

Chris Van Der Lee, Albert Gatt, Emiel Van Miltenburg, Sander Wubben, and Emiel Krahmer. 2019. Best practices for the human evaluation of automatically generated text. In *Proceedings of the 12th International Conference on Natural Language Generation*, pages 355–368.

Jesse Vig and Yonatan Belinkov. 2019. Analyzing the structure of attention in a transformer language model. In *Proceedings of the 2019 ACL Workshop BlackboxNLP: Analyzing and Interpreting Neural Networks for NLP*, pages 63–76.

Oriol Vinyals, Meire Fortunato, and Navdeep Jaitly. 2015. Pointer networks. *Advances in Neural Information Processing Systems*, 28:2692–2700.

Yequan Wang, Minlie Huang, Xiaoyan Zhu, and Li Zhao. 2016. Attention-based LSTM for aspect-level sentiment classification. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 606–615.

Sam Wiseman, Arturs Backurs, and Karl Stratos. 2021. Data-to-text generation by splicing together nearest neighbors. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 4283–4299.

Wei Xu, Courtney Napoles, Ellie Pavlick, Quanze Chen, and Chris Callison-Burch. 2016. Optimizing statistical machine translation for text simplification. *Transactions of the Association for Computational Linguistics*, 4:401–415.

Diyi Yang, Aaron Halfaker, Robert Kraut, and Eduard Hovy. 2017. Identifying semantic edit intentions from revisions in Wikipedia. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 2000–2010.

David Zajic, Bonnie Dorr, and Richard Schwartz. 2002. Automatic headline generation for newspaper stories. In *Workshop on Automatic Summarization*, pages 78–85.

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.

Zhemin Zhu, Delphine Bernhard, and Iryna Gurevych. 2010. A monolingual tree-based translation model for sentence simplification. In *Proceedings of the 23rd International Conference on Computational Linguistics*, pages 1353–1361.

A Classifier Details

Even after filtering by heuristic, we found that many of the remaining headline changes still do not reflect a substantive update to the article. To identify such cases and filter them out, we develop a binary classifier which is trained to determine whether H_1 needs to be updated based on the changes between B_1 and B_2. The class labels correspond to the positive and negative classes that we previously defined. We fit a logistic regression model trained on a set of 1,041 features. Since articles pertaining to certain topics are more likely to evolve (e.g., severe weather), we learn topic vectors using a 200-component non-negative matrix factorization (NMF) of article text.[14] NMF topics are derived from TF-IDF bag of word feature vectors constructed with a vocabulary size 67,950 (corresponding to a minimum document frequency of 10). We partitioned tokens into separate documents based on where they occurred: H_1, B_1, B_2, removed tokens, and added tokens; thus we are able to learn separate NMF topics for each of these modalities. We additionally incorporate 41 features, many of which are derived from prior work in classifying edits in Wikipedia articles (Daxenberger and Gurevych, 2013; Yang et al., 2017). Descriptions of these features are given in Table A.1. We trained this filter by weighting example loss by the inverse class weight, and tuning an L1 regularization penalty on the validation set (tuned for F1 score).[15]

We use a random sample of 10% of the examples belonging to each class for training and evaluation. Our best classifier achieves 51.9 F1 after tuning the regularization penalty. This is a difficult problem, as both the training and evaluation data are silver-labeled, and noisy (i.e., not all revised headlines required revision). Note that we are only learning this model in order to additionally clean the data, and operate under the assumption that this classifier will learn that more extreme body edits warrant a headline update.

During inference, we predict the positive label if the probability is above a threshold of 0.7 and the negative label otherwise – tuned to maximize F1 on the validation set. We compare this model to majority and random classifier baselines (averaged across three runs). Additionally, since there is a strong correlation between the content mentioned in the headline and the lead sentences, we include baselines which predict the positive label if there is a change to the lead-1, lead-3, or lead-5 sentences (Table A.3). In spite of outperforming all baselines on F1 and accuracy, the relatively low F1 score underscores the difficulty of this problem, as there are many reasons why a headline may need to be updated: based on editorial whim, stylistic concern, or other cosmetic changes that are not grounded in a change to the underlying facts of the article.

A.1 Feature Weights

Only 87 out of 1,041 had non-zero weight, a byproduct of training with an L1 regularization penalty. The following features had high positive weight: *Has change in lead-1*, *Has change in lead-5*, *lexical overlap between H_1 and edited body tokens*, *lexical overlap between H_1 and removed tokens in lead-1*, *lexical overlap between H_1 and removed body tokens*, *wire=NY Times*, and *ratio of unique added tokens in the lead-1*. On the other hand, the features which had the most negative weight were: *wire=BBC*, *wire=Guardian*, and CosSim (H_1, B_2).

The majority of these follow our intuition. For example, a change in the lead sentence(s) means a headline update is more likely; if H_1 and B_2 are *not* similar, then H_1 likely needs to be updated to better reflect B_2. With respect to the sources, New York Times headlines tend to be edited more often than other sources, and BBC and The Guardian tend to have more examples with body-only updates. We included the source as a feature to account for the effect of different newsrooms as well as the process used to collect article updates for those particular sources.

We also explored which topics had high positive/negative weights. We find the topic with the highest weight corresponded to {'arrested', 'charged', 'old', 'murder', 'suspicion', 'custody', 'magistrates', 'bail', 'appear', 'aged'} (shown here by the ten tokens in this topic vector with highest weight). This aligns with our observation that headlines for articles tracking criminal investigations are updated frequently in our corpus. The following topic had a large negative weight: {'send', 'comments', 'conditions', 'pictures', '100', 'yourpicsbbccouk', 'terms', 'text', 'upload', 'file'}. This

[14] https://scikit-learn.org/stable/modules/generated/sklearn.decomposition.NMF.html

[15] https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LogisticRegression.html

Feature Description	Text Type	Count
NMF	H_1, B_1, B_2, removed tokens, inserted tokens	1000
Lexical overlap between H_1 and removed tokens	B_1, lead-1, lead-3, lead-5	4
Lexical overlap between H_1 and all edited tokens	B_1 + B_2 $^{\uparrow}$, lead-1$^{\uparrow}$, lead-3, lead-5	4
Has change in lead sentence	lead-1$^{\uparrow}$, lead-3, lead-5$^{\uparrow}$	3
# unique tokens removed / # unique tokens	B_1 + B_2, lead-1, lead-3, lead-5	4
# unique tokens added / # unique tokens	B_1 + B_2, lead-1, lead-3, lead-5	4
Ratio of B_1 to B_2	tokens, token types, sentences, capital letters, punctuation, characters, numbers	7
# tokens for edit type / total # edited tokens	insert$^{\downarrow}$, delete, replace old, or replace new tokens	4
CosSim(B_1, B_2)		1
CosSim(H_1, B_1)		1
CosSim(H_1, B_2)$^{\downarrow}$		1
|CosSim(H_1, B_1) − CosSim(H_1, B_2)|		1
CosSim($NMF(B_1)$, $NMF(B_2)$)		1
News wire (WaPo$^{\downarrow}$, NYT$^{\uparrow}$, Independent, Guardian$^{\downarrow}$, BBC$^{\downarrow}$)		5

Table A.1: Feature sets used to build the classifier. The five features with the largest positive standardized regression coefficients (normalized by standard deviation of associated feature) are indicated by $^{\uparrow}$, and the five most negative are indicated by $^{\downarrow}$. Cosine similarity is computed between TF-IDF weighted bag of word vectors unless otherwise noted.

	Train	Valid	Test	Total
# Examples	75,075	9,385	9,387	93,847
Positive	11,792	1,316	1,350	14,458
Negative	63,283	8,069	8,037	79,389

Table A.2: Descriptive statistics on data used to train/evaluate the classifier used for filtering.

	P	R	F1	Acc
Majority label	0.0	0.0	0.0	85.6
Random	15.3	16.5	15.9	74.8
Change in Lead-1	41.1	55.5	47.2	82.2
Change in Lead-3	30.4	71.7	42.7	72.3
Change in Lead-5	25.6	**78.7**	38.6	64.0
Logistic regression	**53.0**	50.9	**51.9**	**86.4**

Table A.3: Precision, recall, F1, and accuracy on the test set for classifier.

topic represents metadata that is often added or removed in articles which usually have no impact on the headline, since they are unrelated to the article's content. Below, we list the all topics with a **positive** weight, with the specific document type for each indicated in parentheses.

- arrested, charged, old, murder, suspicion, custody, magistrates, bail, appear, aged (added, removed, B_2, B_1)
- officers, officer, ipcc, policing, force, constable, chief, armed, pc, taser (added, B_1)
- maduro, venezuela, chavez, opposition, president, venezuelan, caracas, assembly, capriles, hugo (B_2)
- incident, woman, scene, bst, area, old, anyone, house, injuries, street (added, removed, B_1, H_1)
- israel, israeli, netanyahu, jewish, jerusalem, minister, jews, prime, palestinians, israelis (B_1)
- korea, north, korean, south, pyongyang, missile, jong, seoul, test, sanctions (B_2)
- report, committee, review, found, recommendations, commission, findings, published, concluded, evidence (removed)
- gas, fracking, shale, drilling, cuadrilla, explosion, site, energy, coal, natural (removed)
- ship, coastguard, rescue, boat, search, vessel, crew, helicopter, coast, missing (B_2)
- prices, price, market, house, average, fuel, cost, costs, nationwide, petrol (removed)
- russia, russian, putin, moscow, kremlin, vladimir, russians, sanctions, crimea, soviet (B_1)
- india, indian, modi, delhi, singh, mumbai, kashmir, hindu, gandhi, bjp (B_2)
- burma, suu, kyi, aung, myanmar, san, burmese, nld, military, democracy (B_1)
- assange, embassy, sweden, wikileaks, ecuador, swedish, extradition, julian, asylum, arrest (B_2)
- company, business, firm, executive, companies, chief, shareholders, shares, profit, profits (removed)
- french, france, paris, hollande, sarkozy, mali, calais, president, francois, nicolas (H_1)
- state, islamic, governor, group, officials, department, airstrikes, fighters, official, isil (B_1)
- madeleine, mccann, portuguese, missing, search, portugal, murat, disappearance, luz, praia (B_2)

The topics with a **negative** weight:

- send, comments, conditions, pictures, 100,

	Train			Valid			Test		
	Pos	**Neg**	**Total**	**Pos**	**Neg**	**Total**	**Pos**	**Neg**	**Total**
HREN	57,285	0	57,285	5,769	0	5,769	6,189	0	6,189
Unfiltered *Pos*	103,806	0	103,806	11,048	0	11,048	12,068	0	12,068
Unfiltered *Pos* + *Neg*	103,806	567,502	671,308	11,048	55,605	66,653	12,068	71,477	83,545

Table A.4: Distribution of data, including HREN, unfiltered *positive*, and unfiltered *positive+negative*.

	METEOR	**ROUGE-L**	**BLEU-4**	**GLEU**	**SARI**
HREN (test)					
Unfiltered *Pos* + *Neg* (train)	29.6	49.4	34.5	19.1	16.6
Unfiltered *Pos* (train)	32.5	50.9	35.0	25.0	**39.7**
HREN (train)	**34.0**	**52.4**	**36.5**	**26.0**	**39.7**
Unfiltered *Pos* (test)					
Unfiltered *Pos* + *Neg* (train)	33.7	56.0	40.1	23.1	18.6
Unfiltered *Pos* (train)	35.2	56.1	39.5	26.5	**33.7**
HREN (train)	**36.2**	**57.0**	**40.7**	**27.4**	**33.7**

Table A.5: Evaluating the BART H_1 + B_{edits} (change only) on different training and test configurations. Bolded corresponds to test evaluation set, and rows below correspond to the training data.

yourpicsbbccouk, terms, text, upload, file (added, removed)

- press, associated, copyright, redistributed, rewritten, material, reserved, broadcast, 2016, published (B_1)
- music, festival, band, album, singer, song, show, songs, concert, fans (B_1)
- editors, picks, commentary, inbox, delivered, top, morning, day, news, subscribe (removed, added, B_1)
- gas, fracking, shale, drilling, cuadrilla, explosion, site, energy, coal, natural (H_1)
- storm, hurricane, winds, mph, coast, typhoon, hit, tropical, power, cyclone (B_1)
- dr, research, study, researchers, brain, university, scientists, cells, science, disease (B_2)
- prices, price, market, house, average, fuel, cost, costs, nationwide, petrol (B_1)
- film, films, actor, movie, best, star, award, director, actress, hollywood (B_1)
- cent, pound, poll, sterling, billion, million, survey, since, around, according (B_1)
- immigration, home, immigrants, office, migration, illegal, deportation, asylum, visa, net (B_2)
- park, choi, south, site, festival, parks, parking, national, lee, seoul (B_1)
- refugees, refugee, asylum, seekers, camp, syrians, camps, countries, aid, syrian (B_1)
- bank, rbs, barclays, lloyds, banking, hsbc, customers, england, co, carney (B_1, H_1)
- news, hacking, murdoch, phone, coulson, editor, brooks, world, newspaper, paper (B_2)
- company, business, firm, executive, companies, chief, shareholders, shares, profit, profits (B_2)
- rates, rate, interest, fed, economy, unemployment, reserve, federal, percent, bank (B_2)
- flooding, flood, rain, river, floods, heavy, flooded, homes, environment, weather (B_1)

A.2 Value of Filtering the Train Set

To study the impact of using the classifier to filter the training data, we also compare with training and evaluating on unfiltered data, particularly unfiltered *positive* examples ($H_1 \neq H_2$ but could include cosmetic updates unrelated to body changes) and unfiltered *positive* + *negative* examples (additionally including examples where $H_1 = H_2$). Note that the training, validation, and test sets for these use the same date cutoffs listed in Appendix B. We provide the sizes of these specialized datasets in Table A.4. In Table A.6, we evaluate the effect of training on these three differently filtered datasets.

First, by comparing performance between training on unfiltered *positive+negative* and unfiltered *positive*, especially with respect to edit metrics (GLEU and SARI), we show that a generation model cannot easily differentiate between *positive* and *negative* examples, to identify when to make edits. Now, we compare training on unfiltered *positive* and the filtered *positive* (i.e., HREN) datasets. We find that training on HREN achieves the best headline generation performance overall, even for the unfiltered *positive* test set, highlighting the value of training on this cleaner subset.

We also find that the examples scored positively by the classifier are less likely to correspond to purely stylistic headline rewrites than unfiltered examples. See Table A.7 for headlines of test examples that were passed or rejected by the classifier, versus examples that had *some* textual change to the headline but were not filtered by the classifier.

	Generation					Classification			
	METEOR	ROUGE-L	BLEU-4	GLEU	SARI	P	R	F1	Acc
Unfiltered *Pos* + *Neg* (test)									
HREN (train)	79.9	89.0	83.6	81.2	31.6	26.1	53.4	35.1	71.5
Unfiltered *Pos* (train)	78.2	89.2	83.0	80.5	31.7	23.8	**59.0**	33.9	66.7
Unfiltered *Pos* + *Neg* (train)	87.6	93.0	90.0	87.4	30.9	16.6	9.9	12.4	79.8
Pipeline	**89.2**	**93.2**	**90.3**	**88.1**	**32.5**	**63.6**	36.7	**46.5**	**87.8**

Table A.6: Bolded corresponds to test evaluation set, and rows below correspond to the training data. H_1 + B_{edits} (change only) models.

The RAND examples in Table A.7 correspond to examples that would have been considered positive examples in HREN, if we had not filtered by classifier. Unlike the filtered POS examples, which predominantly correspond to substantive changes in the article, there are several instances of headline changes in RAND that are stylistic in nature.

A.3 Filter & Generate Pipeline Evaluation

Although we primarily use this classifier for filtering, we believe it can be useful in a headline udpating pipeline for determining *when* a headline update is necessary, as not all body changes warrant corresponding headline changes. Here we conduct a preliminary analysis of this. Namely, for the unfiltered *positive+negative* test set, we compare how training on the various configurations from Table A.5 compares to a pipelined approach.

In the pipeline, examples are first passed through the classifier, and if the probability of the positive label is below 0.7, H_1 is simply copied as the prediction for H_2. Otherwise, we use the BART H_1 + B_{edits} (change only) model trained on HREN's training set and take its output as the prediction for H_2.

Pipeline performance is displayed in Table A.6. In addition to the generation metrics described in Section 6, we also evaluate performance with respect to classifying whether H_1 needs to be updated, where we treat a predicted headline that is not identical to H_1 as implicitly predicting the positive label. We find that pipelining outperforms all generation models alone, regardless of the data they were trained on. While training on unfiltered *positive+negative* examples achieves comparable performance on generation metrics, it performs very poorly on classification, as it results in the model not learning to make edits for almost all positive examples.

Human Evaluation on Unfiltered Examples Table A.8 contains the results from the human evaluation on both the Pos only dataset in addition to pipelined systems on the unfiltered test set using difference headline generation models. On the unfiltered test examples, the only significant interaction is for **minimal edits**, and Tukey's HSD identifies both H_1+B_{edits} (change only) and COPY H_1 as being rated higher than B_2 ($p = 3.3e-4$ and $p = 1.1e-2$, respectively). Note that 93% of these examples are negative (i.e., $H_1 = H_2$), and so it is not surprising that there is little difference in how competing model predictions are scored; on this subset, most models copy H_1.

B Time-Based Partitioning

The time frames used for partitioning the classifier data are given below:

- **Train**: 08/29-2006 11:30 - 09/01/2017 19:00
- **Valid**: 09/01/2017 20:00 - 01/24/2019 11:15
- **Test**: 01/24/2019 12:15 - 01/14/2021 23:38

The same date cutoffs are used for partitioning generation task as well, except that the minimum date for the test set is set to March 1, 2019. This is to avoid potential contamination between the data used to pretrain BART (Lewis et al., 2020), the pretrained transformer we finetune in many of our experiments. BART pretraining data includes CC-News[16] articles crawled between September 2016 and February 2019. The specific version of BART we use in our work was originally fine-tuned for summarization on CNN-Daily Mail, consisting of news articles before April 2015. Therefore, we do not expect there to be any overlap with our test set, in terms of stories tracking the same events.

C Human Evaluation Design

Annotators were presented with a *diff* between B_1 and B_2 along with H_1, and they were asked to judge a candidate updated headline on five dimensions using a Likert scale. Following established guidelines in evaluating generated text (Van Der Lee et al., 2019), the first two dimensions correspond to whether the candidate headline is **factual**

[16] https://huggingface.co/datasets/cc_news

Score	Old Headline	New Headline
POS	Three hurt as car strikes buffalo	Man rescued as car hits buffalo
	Syria conflict: Peace talks due to begin in Astana, Kazakhstan	Syria conflict: Peace talks begin in Astana, Kazakhstan
	Israeli woman killed as Palestinian stabbings add to escalating violence	Israeli woman and soldier killed in Palestinian stabbings
	Security alert after cash raid	Cash box found after Lisburn raid
	Santander profits hit by higher PPI compensation	Santander confirms profits hit by PPI compensation
NEG	UN Security Council 'failing Syrian people'	Syria crisis: UN Security Council 'failing victims'
	Ikea relaunches furniture recall after child dies	Ikea US relaunches furniture recall after child dies
	Mali's Festival au Désert cancelled amid fears of extremist violence	Mali cancels return of famous music festival after al-Qaida attack
	European governments refuse to follow Trump on status of Jerusalem	Europe tells Netanyahu it rejects Trump's Jerusalem move
	Graves exhumed in hunt for missing mother Natalie Putt	Graves dug up in hunt for missing mother Natalie Putt
RAND	Chile tycoon 'wins' first round	Chilean tycoon wins first round
	Helen Bailey murder detective charged with stealing £9,000	Helen Bailey murder detective charged with stealing £9,000 from a safe
	HSBC shares down as full year profit falls 62%	HSBC shares down as annual profit falls 62%
	Paddy Power's Oscar Pistorius ad to be withdrawn with immediate effect	Paddy Power's Oscar Pistorius ad to be pulled after record 5,200 complaints
	Bank of England keeps interest rates on hold	Pound jumps as Bank of England hints at rate rise

Table A.7: Headlines from the classifier test set containing some textual change between version pairs, which were either scored POSitive or NEGative by the classifier filter, or were sampled at RANDom prior to filtering by the classifier. Headline pairs scored as NEGative by the filter tend to contain purely stylistic changes, as do RAND.

and **grammatical**. Based on the intuition that as few changes as possible should be made to the original text for such editing tasks (Dahlmeier et al., 2013), our third dimension corresponds to **minimal edits**. Next, given that our task pertains to updating headlines for evolving news stories, we want to ensure that the candidate headline focuses on the important changes/information in the updated version of the article, which we refer to as **focus**. Finally, since the structure and phrasing of headlines often deviate from other forms of text (Straumann, 1935; Mårdh, 1980), we aim to evaluate whether the candidate headline is brief and uses language the way that a typical headline would. We call this last dimension **headlinese**.

We select 200 examples for this study, with 143 being randomly sampled from HREN. The remaining 57 are randomly sampled from 83,545 *unfiltered* examples, consisting of 14.4% positive and 85.6% negative examples which fall within the same date ranges of HREN's test set (and could possibly have overlap with HREN as well). Because we trained generation models only on the filtered training set, during inference we pipelined these generation models with our classifier, copying the H_1 if the example did not meet the threshold of 0.7, otherwise the headline predicted by the generation model was used. We report results separately for the two test sets.

Each candidate headline was completed by 3 unique annotators. All annotators were native English speakers familiar with news headlines from major US and UK papers, and two received degrees in journalism. Annotators were financially compensated with an hourly rate above the minimum wage for their location. When more than one model

		Factual†		Focus†		Min Edits†		Headlinese		Grammatical*	
		Avg	%Dis	Avg	%Dis	Avg	%Dis	Avg	%Dis	Avg	%Dis
HREN (143)	COPY H_1	4.627	9.8	4.263	18.4	**4.956**$^{\sharp\natural}$	1.2	4.972	0.5	**4.998**	0.0
	B_2	4.881$^{\flat}$	3.0	4.669$^{\flat}$	6.3	1.855	87.9	4.956	1.2	4.965	0.9
	H_1 + B_2	4.904$^{\flat}$	2.1	**4.706**$^{\flat}$	5.6	3.145$^{\natural\sharp}$	49.7	**4.981**	0.2	4.951	1.4
	H_1 + B_{ed} (ch only)	4.807$^{\flat}$	5.6	4.639$^{\flat}$	7.7	3.354$^{\natural\sharp}$	44.1	4.960	0.9	4.953	0.9
	Gold	**4.916**$^{\flat}$	1.6	**4.706**$^{\flat}$	4.4	2.301$^{\natural}$	74.8	4.963	0.7	4.984	0.2
		Factual		**Focus**		**Min Edits***		**Headlinese**		**Grammatical**	
Unfiltered (57)	Copy H_1	4.860	4.7	**4.749**	5.3	**4.965**$^{\natural}$	1.2	**4.947**	2.3	**4.994**	0.0
	B_2	4.877	4.7	4.731	7.6	4.649	8.8	4.901	4.1	4.982	0.6
	H_1 + B_2	**4.901**	4.1	4.719	7.0	4.807	4.7	4.918	3.5	**4.994**	0.0
	H_1 + B_{ed} (ch only)	**4.901**	4.1	4.743	7.0	4.895$^{\natural}$	2.9	4.918	3.5	**4.994**	0.0
	Gold	4.860	4.7	4.743	5.8	4.760	7.0	**4.947**	2.3	**4.994**	0.0

Table A.8: Test. Differences that are statistically significant by Tukey HSD at the $p < 0.05$ level for HREN are indicated by superscripts. Superscripts indicate that the model is significantly better than COPY H_1 $^{\flat}$, Gold$^{\sharp}$, or B_2 $^{\natural}$. Best average score for each item is in **bold**. ANOVA statistical significance level is indicated on the column header ($* : p < 0.05$, $\dagger : p < 10^{-10}$). *%Dis* corresponds to the % of annotations where an annotator did not agree with the Likert item (score < 4).

made identical predictions, we attributed each annotator's judgments to all models that would have generated that prediction.

To avoid using untrained small batch annotations for human evaluation of NLG models (Clark et al., 2021), we worked closely with annotators and engaged in a round of remediation on 75 examples prior to running this study, to make sure the instructions were clear. Note that our task is grounded in an actual news article, making it easier to discriminate between clearly misleading/false headlines, rather than tasks where a model can freely draft text (e.g., draft a work of fiction).

Inter-Annotator Agreement Table C.9 displays the % agreement between annotators for each item. There was a strong bias toward scoring most items with 5 (*Strongly agree*), which partially drives the strong agreement rates. This is underscored by lower correlation coefficients between annotators (Table C.10). **Grammatical** and **headlinese** have low correlation coefficients as there is near unanimous agreement for this item, with only a few examples available to provide signal for ranking. For example, only 16 of 3000 unique annotations were scored lower than 5 for **grammatical**. Conversely, annotators achieved relatively low inter-annotator agreement on **minimal edits**, but achieve high rank correlation.

Statistical Significance In order to test for statistical significance, we ran multi-way ANOVAs for each Likert item with headline prediction *model* (gold, Copy H_1, B_1, H_1+B_2, H_1+B_{edits} (change only)), *example ID*, and *annotator ID* as independent variables. Separate tests were run for the unfiltered and the positive only (filtered) test sets. If a statistically significant effect was found for *model* at the $p < 0.05$ level, we ran Tukey's post-hoc HSD test to identify which models tended to be rated differently from each other.

D Human Evaluation Guidelines

Figure D.1 displays an example of the task interface for the human evaluation. Below are the exact guidelines provided to annotators, as recommended by Schoch et al. (2020).

Overview

News articles are often updated after they are published online. When facts are corrected, or new facts added to the news article, the headline may also need to be updated to reflect those changes. In this task, you will be shown an original English news article, the original headline, and all revisions made to the article body. Given a headline for the revised article, your task is to mark how strongly you agree or disagree with whether the updated headline:

- Is factually correct
- Is free of typos and is grammatical
- Focuses on important changes/information in the updated version
- Makes as few changes as possible to the original headline
- Is brief and uses language the way that a typical headline would; looks like "headlinese"

	Raw			Binned		
	1 & 2	**1 & 3**	**2 & 3**	**1 & 2**	**1 & 3**	**2 & 3**
factual	86.5	90.1	83.6	93.9	92.8	93.2
focus	70.0	56.7	56.1	89.3	84.0	88.0
minimal edits	65.1	53.3	56.0	89.2	83.1	82.3
grammatical	98.5	95.8	96.7	99.0	98.6	98.9
headlinese	96.5	95.0	93.5	98.3	97.8	98.0

Table C.9: Percent inter-annotator agreement between each pair of annotators (column) for each Likert item (row). Agreement is computed over raw response, and after binning responses into *Not Agree* (< 4) vs. *Agree* (≥ 4).

	1 & 2		**1 & 3**		**2 & 3**	
factual	0.266	(1.2e-14)	0.135	(9.3e-5)	0.180	(1.4e-7)
focus	0.174	(2.2e-7)	0.027	(0.44)	0.156	(8.0e-6)
minimal edits	0.751	(9.5e-131)	0.712	(5.3e-119)	0.651	(1.1e-106)
grammatical	-0.006	(0.87)	0.047	(0.18)	-0.009	(0.80)
headlinese	0.057	(0.10)	0.096	(6.3e-3)	0.047	(0.18)

Table C.10: Kendall's tau rank correlation coefficient between each pair of annotators (column) for each Likert item (row). Correlation coefficient is computed using raw responses. P-value is indicated in parentheses.

Steps

1. Read through the original news article and headline on the lefthand side, and the revised article body on the righthand side. Pay particular attention to what revisions were made to the article, as well as the content in the original article. Text that was removed from the original article will be highlighted in red, text that was added to the revised article will be highlighted in green, and substitutions will be highlighted in yellow.
2. After reading the original and revised news articles, consider the candidate headline for the revised article and rate how strongly you agree with the statements:
   (a) **Is factually correct**. A headline should never state facts that are not supported by the body of the news article, either extrapolations or clearly contradicting the news body.
   (b) **Is free of typos and is grammatical**. A good headline should not contain typographical errors or clear grammatical mistakes.
   (c) **Focuses on important changes/information in the updated version**. If there are any critical changes to the new version of the story, the headline should highlight these changes. If only minor changes were made to the article, then the new headline should focus on the important information in the article overall.
   (d) **Makes as few changes as possible to the original headline**. It is also important that the new headline preserves the structure of the original headline as much as possible. In other words, a good revised headline should make as few edits to the original headline as possible. This is most important if there were only minor changes to the article.
   (e) **Is brief and uses language the way that a typical headline would; looks like "headlinese"**. English headlines are written in a unique form of language called "headlinese". Some hallmarks of headlinese are omission of articles like "a" or "the", constructions like "Parliament to pass bill" for an event that is expected to occur in the future, and generally keeping headline as short as possible. A good headline should look like it is written in headlinese.
3. You should judge how strongly you agree with each of the above statements on a scale of Strongly disagree, if the statement is certainly wrong, to Strongly agree, if you are sure it is correct.
4. Write any additional comments or questions about the example in the comment box. You should use the comment box if you think this example is malformed or there were problems in processing, if there is a problem with the headline that isn't captured by your judg-

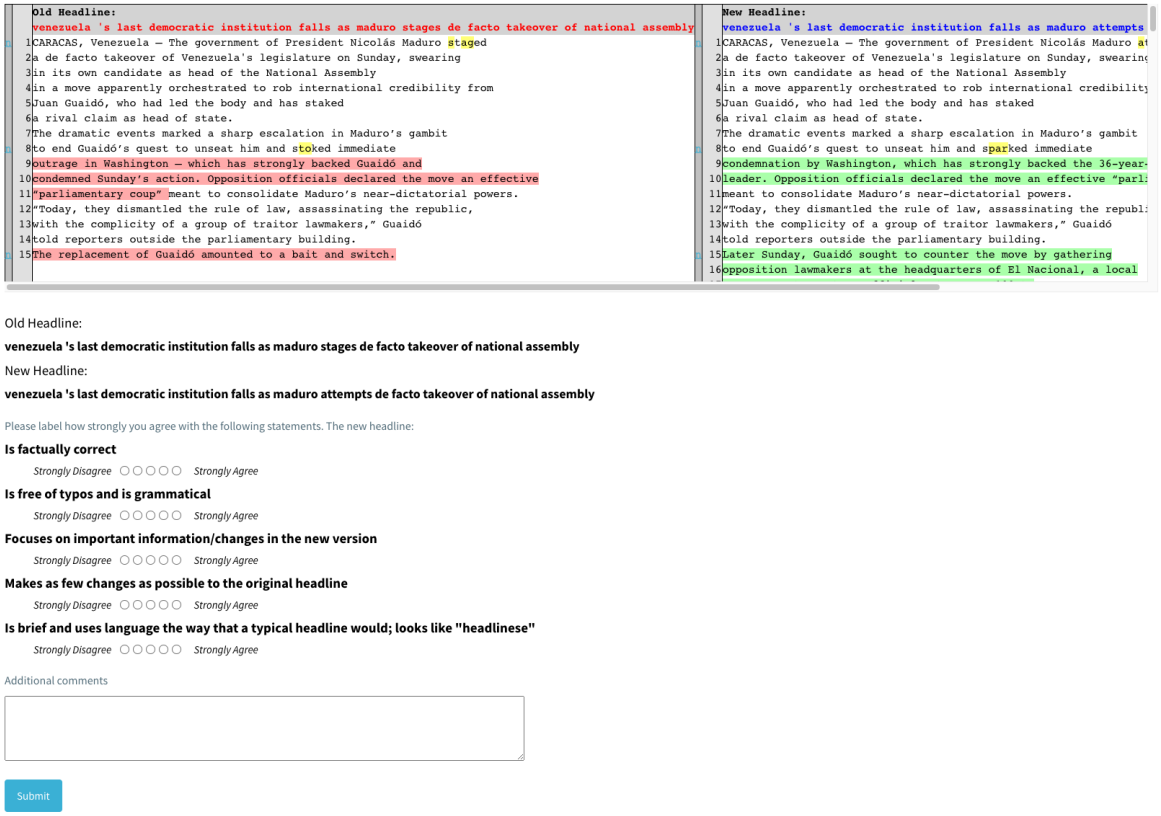

Figure D.1: Example task used in the human evaluation.

ments, if you are uncertain about your judgments, or for any other reason.

5. Click the **Submit** button in order to record your choices and move on to the next task.

Tips

- You should not try to find these articles using a search engine. Make your decisions based only on the information presented in the task. This is especially important since these particular news articles you will annotate evolved over time, and different versions may have different headlines.
- For most examples, you do not need to read both versions of the article in detail. Reading the first few paragraphs and looking at what changed between versions is usually sufficient to judge the revised headline. Annotating examples in this way is perfectly acceptable.
- The fact that the candidate headline is lowercased and tokenized should not influence your judgment. All headlines are lowercased, split up into words/punctuation, and then separated by spaces as part of our preprocessing.
- Only a small portion of the news article will be visible, so that you do not need to scroll very far to view the questions. You can scroll within the story box to view the rest of the article.
- You may see the same example with a similar candidate headline you annotated before. This is not an error, but rather, the prediction of a model that happened to be similar to one before.
- Use the **Comments** box for any additional comments.
- You should not use Internet Explorer for completing these tasks.

E BART Attention

Which tokens are most important for headline rewriting? Understanding which words are attended to by a neural network with multiple layers of multi-headed attention is, needless to say, difficult. Here we follow the method proposed in Vig and Belinkov (2019) and aggregate attention across classes of context token types to reduce the number of attention distributions. We investigate the strongest headline rewriting model in all analyses, the $H_1 + B_{edits}$ (change only) BART model, and only consider the attention heads for the decoder network. We focus solely on the decoder network as we would like to determine which sections of the context were attended to most by the network

as different tokens are (greedily) decoded.

We label all tokens in the context by the span they occur in. A token can either belong to the **Headline**, or a **Delete** (removed without replacement from B_1), **Replace-Old** (replaced token from B_1), **Replace-New** (substitute token added in B_2), or **Insert** (B_2 token without analogue in B_1) edit span. Because this model operated only on body text that differed between article versions, we have no tokens which were present in both the old and new article bodies. Special tokens indicating the start or end of different spans were assigned the **Other** type to ensure a valid probability distribution across span types.

Corpus-level Analysis We first investigate whether particular layers/heads are biased towards particular span types. We compute the average attention placed by a head, α, across the entire corpus for a particular `span` type by:

$$P_\alpha(\texttt{span}) = \frac{\sum_{c,h \in X} \sum_{i=1}^{|c|} \sum_{j=1}^{|h|} \alpha_{i,j} \mathbb{1}[c_i = \texttt{span}]}{\sum_{c,h \in X} \sum_{i=1}^{|c|} \sum_{j=1}^{|h|} \sum_{k=1}^{|S|} \alpha_{i,j} \mathbb{1}[c_i = S_k]} \quad (1)$$

where X is the set of examples, c is the list of span types for each context token, h is the list of tokens in H_2, S is the list of span types, and $\alpha_{i,j}$ is the attention placed on context token i while decoding token j for H_2.

Figure E.2 shows the mean attention paid to different span types. Headlines and added content – Replace-New and Insert – are heavily attended to, whereas removed tokens are less important to the BART decoder. This trend is even more pronounced if we plot the difference between mean attention and what one would expect from an attention head that paid attention to every token equally (Figure E.3). Across the validation set, only 1.5% of tokens are Headline, 5.8% Delete, 10.6% Replace-Old, 19.8% Replace-New, 12.3% Insert, and 49.9% Other.

In addition, the amount of attention paid to added content and headline tokens increase in later layers at the expense of Other tokens (Figure E.4). We posit that this is because the initial layers need to attend to special tag tokens in order to understand the span type of enclosed tokens. It may also just arise from the fact that initially the decoder attends to all tokens relatively uniformly (49.9% of tokens are Other on average). Even in the initial layers though, H_1 tokens are attended to more than would be expected by a uniform attention distribution, likely because that text always appears near the start of the context, and thus identifying enclosing tags is less critical.

H_1	Seventeen dead after plane repatriating Indians stranded by Covid crashes
H_2 (gold)	Eighteen dead after plane repatriating Indians stranded by Covid crashes

Input	**Prediction**
H_1	sixteen dead after plane repatriates indians stranded by covid crashes
B_2	air india express plane skids off runway and breaks in two in kerala
H_1 + B_2	eighteen dead after plane repatriating indians stranded by covid crashes breaks in heavy rain
H_1 + B_{edits}	eighteen dead after plane repatriating indians stranded by covid crashes in heavy rain
H_1 + B_{edits} (change only)	eighteen dead after plan repatriating indians stranded by covid crashes

Table F.11: Predictions for BART under different input representations for https://www.newssniffer.co.uk/articles/1983637/.

Attention Anecdotes We also plot the average attention paid by the decoder to different span types while decoding individual examples. In this case, we aggregate attention over a fixed context, c, for decoded token j by:

$$P_j(\texttt{span}) = \frac{\sum_{i=1}^{|c|} \sum_{\alpha \in A} \alpha_{i,j} \mathbb{1}[c_i = \texttt{span}]}{\sum_{i=1}^{|c|} \sum_{\alpha \in A} \sum_{k=1}^{|S|} \alpha_{i,j} \mathbb{1}[c_i = S_k]} \quad (2)$$

where A is the set of 192 attention heads in the BART decoder, α corresponds to a single decoder attention head, and $\alpha_{i,j}$ is the attention paid to token i while decoding token j in H_2. Figure E.5 displays a handful of examples where the decoder is attending to tokens that were either copied from H_1, or new replacement tokens sourced from B_2.

F Sample Output

We provide sample output from BART generation models in Tables F.11-F.13.

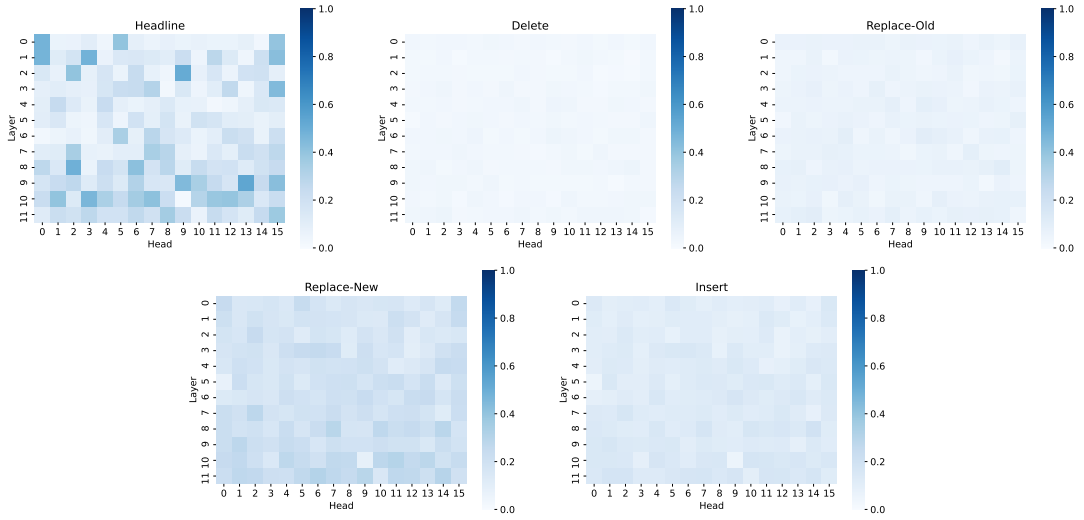

Figure E.2: Mean attention paid to tokens of each span type for individual attention heads in the BART decoder network. Span type is indicated by title above each heat map.

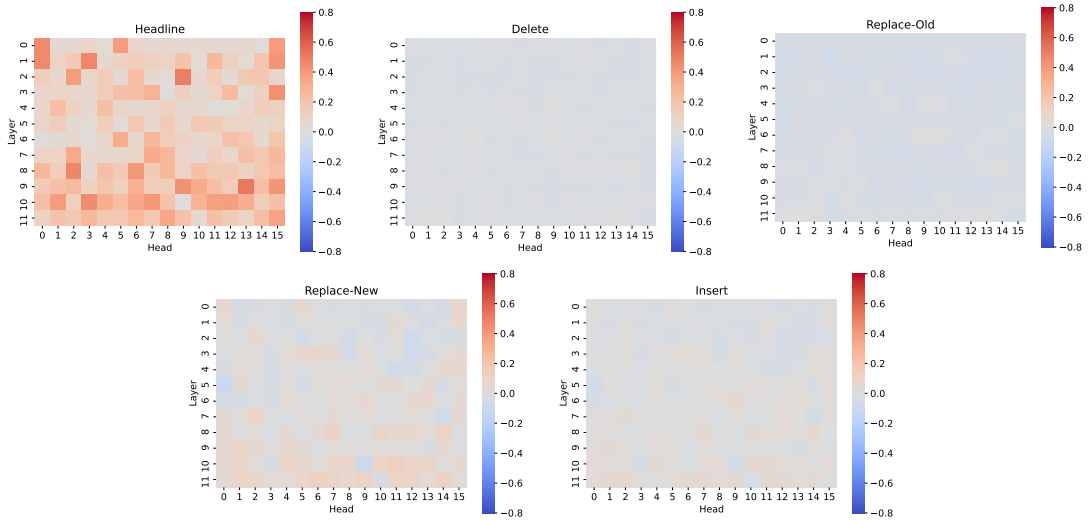

Figure E.3: Mean attention paid to tokens of each span type for individual attention heads in the BART decoder network, relative to that expected under a uniform attention model.

G Reproducibility Checklist

We supplement details provided in the main paper regarding aspects of the reproducibility checklist. We provide automated metrics on the validation set in Table F.14. For pointer networks, we select hyperparameters based on random search. We explored values for dropout between 0.0 and 0.8, learning rate between 10^{-5} and 10^{-3}, number of encoder layers {2, 3}, number of decoder layers {1,2,3}, and hidden dimension {32, 64, 128, 256}. After 8 such configurations, we select the best ones based on performance on the validation data: dropout rate = 0.333, learning rate = 0.00099, encoder layers = 2, decoder layers = 3, hidden size = 64. We use a batch size of 16. We specifically use *glove.6B.300d.txt* GloVe vectors to initialize embeddings. For BART models, we use a batch size of 8, 50 warm-up steps, a weight decay of 0.01, and we fine-tune up to 5 epochs (with fine-tuning taking approximately 6 hours on a single NVIDIA V100 GPU). During inference, we use beam search with beam width = 20 for both models, after considering all values between 1 and 20. We use 20 as it achieved the best performance on the validation set while completing in reasonable time.

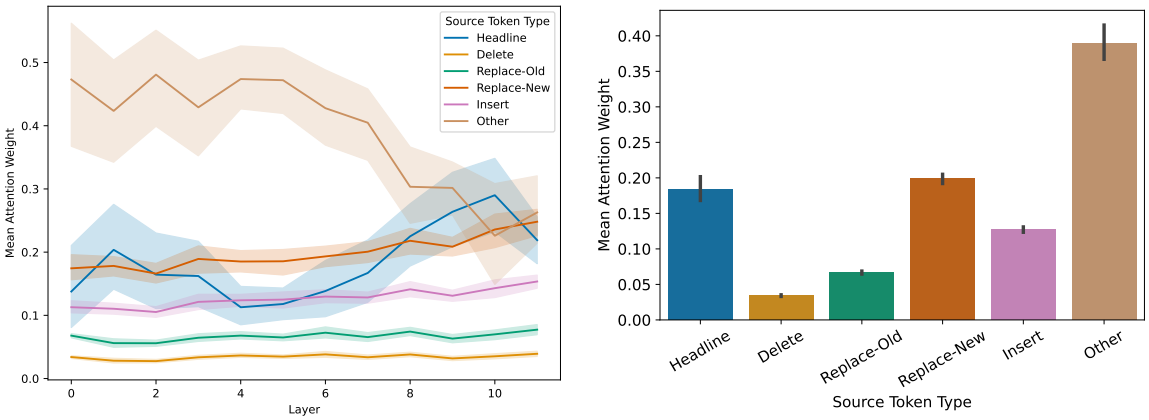

Figure E.4: Mean and 95% confidence interval of attention across layers (left) and overall (right).

H_1	Missing Nora Quoirin: Searchers to play relative's voice recording
H_2 (gold)	Missing Nora Quoirin: Mum's voice used in search

Input	**Prediction**
H_1	missing nora quoirin: norway police release video of relative's voice recording
B_2	malaysian police search for missing teenager
H_1 + B_2	missing nora quoirin: searchers use mother's voice recording
H_1 + B_{edits}	missing nora quoirin: searchers to play mother's voice recording
H_1 + B_{edits} (change only)	missing nora quoirin: search teams play mother's voice recording

Table F.12: Predictions for BART under different input representations for https://www.newssniffer.co.uk/articles/1818912/.

H_1	Global shares up as recovery hopes overshadow virus worries
H_2 (gold)	Wall Street up as recovery hopes overshadow virus worries

Input	**Prediction**
H_1	global shares rise as recovery hopes overshadow virus worries
B_2	us shares open sharply higher on hopes of economic recovery
H_1 + B_2	wall street opens at 3 - month high on recovery hopes
H_1 + B_{edits}	wall street opens sharply higher as recovery hopes overshadow virus worries
H_1 + B_{edits} (change only)	us shares up sharply as recovery hopes overshadow virus worries

Table F.13: Predictions for BART under different input representations for https://www.newssniffer.co.uk/articles/1950357/.

		METEOR	ROUGE-L	BLEU-4	GLEU	SARI
Rule-Based	COPY H_1	30.7	51.0	36.4	20.8	14.9
	LEAD-1	22.2	24.9	8.7	7.2	33.3
	SUBSTITUTION	32.2	51.3	37.1	22.9	20.0
Pointer Network	H_1	26.7	47.2	31.9	20.1	26.6
	B_2	16.0	30.6	16.1	15.7	31.6
	H_1 + B_2	29.8	50.2	34.7	22.4	28.2
	H_1 + B_2 + B_1	29.5	49.8	34.8	21.2	23.9
	H_1 + B_{edits}	31.0	51.0	35.7	23.3	31.5
	H_1 + B_{edits} (change only)	30.2	50.7	35.1	23.0	32.2
BART	H_1	30.0	49.9	35.1	21.2	22.1
	B_2	23.5	38.4	22.0	19.2	36.8
	H_1 + B_2	35.4	54.4	38.2	28.4	**42.9**
	H_1 + B_{edits}	34.6	53.0	37.2	26.4	37.6
	H_1 + B_{edits} (change only)	**36.5**	**54.8**	**39.3**	**29.4**	**42.9**

Table F.14: Automated metrics on the validation set.

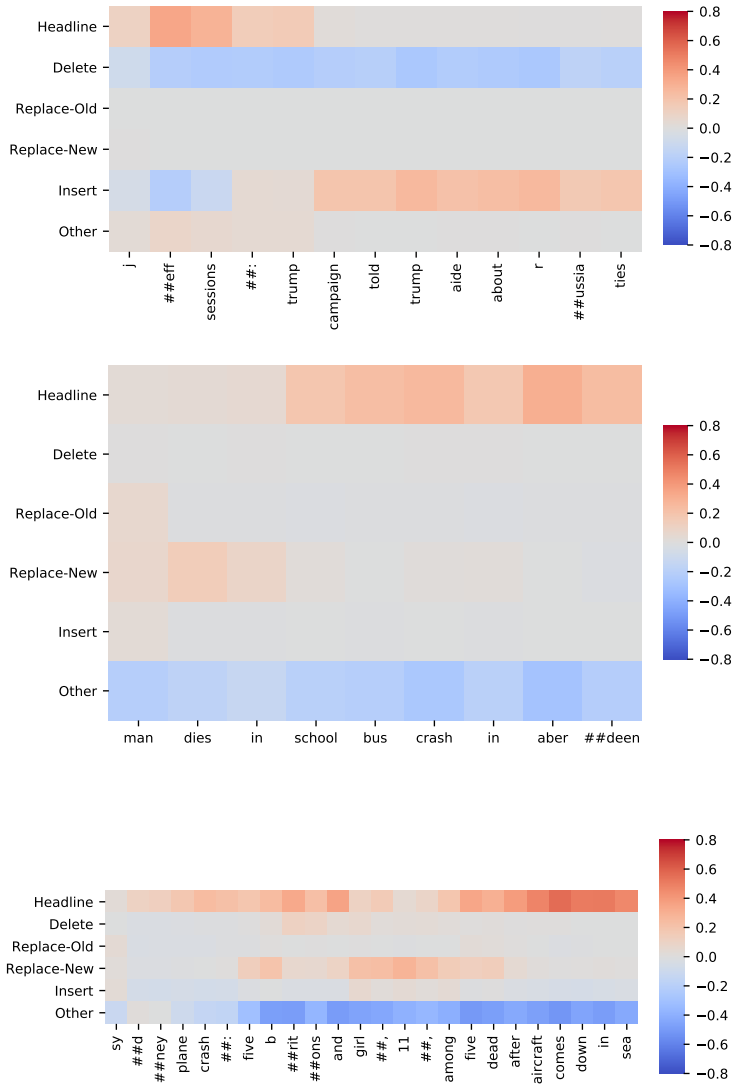

Figure E.5: Mean attention placed on each span type during decoding of H_2 relative to that expected under a uniform attention distribution. Words in the decoded headline are on the x-axis, while the y-axis corresponds to the different types of context tokens. Sources: https://www.newssniffer.co.uk/articles/1493398/, https://www.newssniffer.co.uk/articles/1513150, https://www.newssniffer.co.uk/articles/1521230.