Automatically Discarding Straplines to Improve Data Quality for Abstractive News Summarization

Amr Keleg*, **Matthias Lindemann***, **Danyang Liu***, **Wanqiu Long***, **Bonnie L. Webber**
Institute for Language, Cognition and Computation, University of Edinburgh
{a.keleg,m.m.lindemann}@sms.ed.ac.uk,
{dyliu,wanqiu.long,bonnie.webber}@ed.ac.uk

Abstract

Recent improvements in automatic news summarization fundamentally rely on large corpora of news articles and their summaries. These corpora are often constructed by scraping news websites, which results in including not only summaries but also other kinds of texts. Apart from more generic noise, we identify straplines as a form of text scraped from news websites that commonly turn out not to be summaries. The presence of these non-summaries threatens the validity of scraped corpora as benchmarks for news summarization. We have annotated extracts from two news sources that form part of the Newsroom corpus (Grusky et al., 2018), labeling those which were straplines, those which were summaries, and those which were both. We present a rule-based strapline detection method that achieves good performance on a manually annotated test set[1]. Automatic evaluation indicates that removing straplines and noise from the training data of a news summarizer results in higher quality summaries, with improvements as high as 7 points ROUGE score.

1 Introduction

Automatic text summarization is a challenging task. Recent progress has been driven by benchmarks that were collected by scraping a large collection of web-pages, including Gigaword (Rush et al., 2015), CNN/DailyMail (Nallapati et al., 2016), Newsroom (Grusky et al., 2018), and XSum (Narayan-Chen et al., 2019). Due to the way they are collected, these datasets contain a substantial portion of articles that are paired with texts that are not summaries. This flaw in data quality negatively impacts research in two ways: (i) models trained on these benchmarks tend to reproduce flaws in the data, making them less useful for summarization, and (ii) any evaluation against a reference text is meaningless if the reference is not actually a summary.


INNOVATION

4 Reasons Elon Musk's Hyperloop Could Tank

Don't expect to be riding one by 2020.

By Matt Peckham | Aug. 13, 2013

Figure 1: A **strapline** ("Don't expect ...") that is mistaken for a summary in the Newsroom corpus.

In this work, we present methods for improving the data quality in scraped news summarization corpora, focusing on the Newsroom benchmark (Grusky et al., 2018). We identify two main issues with the data quality: (i) noise in the extraction process (the wrong field being scraped, markup, ...), which was previously also identified to be an issue by Kryscinski et al. (2019), and (ii) *straplines*. According to the writing guidelines used by CERN[2], "[t]he strap[line] gives added "teaser information not included in the headline, providing a succinct summary of the most important points of the article. It tells the reader what to expect, and invites them to find out more."

Figure 1 shows an example of a strapline ("Don't expect to be riding one by 2020") below the regular headline. While the CERN guidelines emphasize the function of straplines to provide a summary, we find that most straplines in the Newsroom corpus are not summaries of their associated articles. Therefore, in order to obtain high quality data, it is necessary to distinguish a strapline aimed at piquing a reader's interest from an abstractive summary. To the best of our knowledge, no work has tried to distinguish straplines from summaries before, and even the word "strapline" does not appear in the ACL anthology in a research paper.

In our work, one pair of us designed a strapline

*Equal contribution

[1]We release our code at https://github.com/nam ednil/straplines

[2]https://writing-guidelines.web.cern.ch/entries/strapline-strap.html

Proceedings of NLP Power! The First Workshop on Efficient Benchmarking in NLP, pages 42 - 51

annotation guideline through discussions and manual pre-annotations (§3.1) and then annotated a development and test set for evaluating strapline classifiers. Based on the guideline, a separate pair created heuristics for a rule-based classifier that distinguishes straplines from summaries (§3.2). We empirically verify the usefulness of these heuristics for strapline detection (§4.2). Automatic evaluation indicates that removing straplines and noise from the training data with our heuristics results in higher quality summaries, with improvements as high as 7 points ROUGE score when compared to reference summaries (§4.3).

2 Related work

Several works have analyzed existing summarization datasets from different aspects but none have identified straplines as an issue. Kryscinski et al. (2019) quantified HTML artifacts in two large scraped summarization datasets which are CNN/DM (Nallapati et al., 2016), and Newsroom (Grusky et al., 2018). They found that "summaries" containing such artifacts were found in $\approx 3.2\%$ of the Newsroom data. They also argued that many of these artifacts could be detected using simple regular expressions and heuristics. Jung et al. (2019) define three sub-aspects of text summarization and analyze how different domains of summarization dataset are biased to these aspects. Bommasani and Cardie (2020) evaluate the quality of ten summarization datasets, and their results show that in most summarization datasets there are a sizable number of low quality examples and that their metrics can detect generically low quality examples. Tejaswin et al. (2021) analyzed 600 samples from three popular datasets, studying the data quality issues and varying degrees of sample complexity, and their analysis of summarization models demonstrate that performance is heavily dependent on the data and that better quality summarization datasets are necessary.

Given that research has shown that the training data of summarization models are noisy, researchers have proposed methods for training summarization models based on noisy data. For example, Kano et al. (2021) propose a model that can quantify noise to train summarization models from noisy data. The improvement of the models indicates that the noisy data has noticeable impacts for the training of the models.

3 Methodology

The Newsroom corpus contains articles from 38 news sources that vary in style and topics. News articles were scraped from HTML pages, where the page's title tag is parsed as the article's headline, while the page's body tag is parsed as the article's body. Since there was no consistent metadata tag indicating the summary of an article, Grusky et al. (2018) used different metadata tags to extract summaries. These tags are generally added to be used by social media platforms, and search engines. News publishers do not share a single format for organizing metadata. Nevertheless, all (or most) use the metadata label *description*, albeit for different things. Since the creators of Newsroom take as the summary of each article, the first tag in its metadata having the keyword *description*, this might be one reason that a strapline appears in the extract for an article in place of the real summary. Knowing that the "summaries" in the Newsroom corpus are of mixed quality, we call what Grusky et al. (2018) scraped from the web *extracts*, which may or may not be a genuine summary.

Grusky et al. (2018) classify extracts according to how much text they repeat verbatim from the article into three categories: extractive (nearly everything appears verbatim in the article), abstractive (summarize in different words) and mixed.

We have focused on extracts classified as "abstractive". We have also limited our study to two of the 38 news sources – ones with different styles and covering different topics, specifically the New York Times (NYT) and time.com.

3.1 Annotation

The extracts in the Newsroom corpus do not all fall neatly into the categories straplines and summaries and noise; in particular, straplines and summaries are not mutually exclusive, and can be seen to form a continuum.

Even in this continuum, what one would definitively classify as a summary depends on multiple factors like its purpose and audience (Spärck Jones, 1999). Therefore, we only identify *common characteristics* of straplines and summaries, restricted to the context of news articles, such as those in the Newsroom corpus. Regarding purpose and audience, we generally assume the audience consists of people who read news on a somewhat regular basis, and that this is the same audience as for the summaries. The purpose is to

provide a brief overview of the news of the day, and we assume this overview includes the headline. This means that the headline plays a central role in our annotation procedure. A practical implication of this is that annotation decisions can sometimes be made very swiftly without reading the actual article.

We identify the following main characteristics of **straplines** that we want to exclude (ordered by importance):

Clickbait A strapline can be designed to attract a reader's attention, rather than being informative.

Little or redundant information A strapline does not add much information to the headline.

General A strapline can make a very general statement, i.e. it would fit for a number of very different articles.

Comment A strapline can be a comment on the event described in the article. This does not apply if the article itself is an opinion piece.

Joke A strapline can be a joke.

Informal A strapline may use informal language.

An extract need not have all the stated properties to be considered a strapline. The characteristics are illustrated in Table 1.

The characteristics of summaries are partially complementary to those of straplines. Again, an extract need not have all the characteristics to be considered a **summary**:

Adds information A summary adds information to the headline.

Relevance A summary contains no irrelevant information and little background information.

Focus The summary of an article describing an event (entity) focuses on that event (entity).

Proposition A summary tends to be one or more propositions.

The following example illustrates that some extracts have characteristics of both a summary and a strapline:

> Jan. 18 Internet Blackout to Protest SOPA: Reddit Says Yes
>
> Following speculation, Reddit has confirmed plans to go dark on Jan. 18 to protest the Stop Online Piracy Act. Wikipedia may follow suit, but what about Google, Facebook and other big-name tech companies?

While the extract adds relevant information to the headline, it also uses a question to attract the reader's attention instead of giving away that "[...] Google and Twitter declined to comment on their support for an Internet blackout", as can be found in the main article.

Labels Because of this overlap in the categories, we annotate each article with one of the following labels: "summary", "strapline", "strapline and summary", "neither" and "paraphrase". We use the category "neither" for noise or when the headline or the extract are difficult to understand before reading the article. We sometimes observe that the extract is a close paraphrase of the headline. By definition, a paraphrase does not add information and therefore would not qualify as a summary. In another use case however, where we assume that a user does not have access to the headline, the extract may provide valuable information. In order to make our annotation more robust to this use case, we include the category of paraphrase, so that those extracts can be included or excluded accordingly.

3.2 Strapline detection pipeline

Before detecting straplines, we preprocess the data to exclude *noisy* extracts (e.g., extracts with HTML tags). Afterwards, the strapline detection method is used to split the remaining extracts into straplines and summaries. The following subsections describe the main heuristics used for noise filtration and strapline detection, with implementation details included in Appendix A.

3.2.1 Noise filtration

Kryscinski et al. (2019) mention that noisy samples represent about 3.2% of Newsroom, hinting that such samples can be detected with simple patterns, but without explicitly describing these patterns. Consequently, we start by looking for patterns of noise in the Newsroom dataset as a first preprocessing step, and identify five clear patterns

Headline	Extract	Characteristic	Heuristic
Awesome! Interactive Internet health map checks your states connection	Check to see if you're part of a bigger problem	Clickbait	Imperative, pronouns
Sochi Olympics: USA Canada Hockey Game Sparks "Loser Keeps Bieber" Ad	USA! USA! USA!	Little information	Too short, exclamation mark
Bill OReilly: More trouble overseas for President Obama and America	The OReilly Factor on FoxNews.com with Bill OReilly, Weeknights at 8 PM and 11 PM EST	General statement	Repeated extract
Sofia Vergara and fiance split, read (and love) the charming statement	At least we know Sofia is probably writing this herself!	Comment	Pronouns
£Quieres seguir viendo noticias en Facebook? Aquí te decimos qué hacer	Facebook cambió su algoritmo para priorizar [...]	N/A	Non-English article

Table 1: Examples of straplines from the Newsroom along with a salient characteristic and the relevant automatic heuristics for strapline detection.

of noise:

Web formatting syntax An extract containing remnants of web formatting syntax. The formatting attributes are inconsistent and not sufficiently relevant for summarization.

Truncation An extract ending abruptly, forming an incomplete sentence. This might be attributed to the fact that news providers tend to have a truncated version of the summary that ended up being scraped in place of the long version of the summary.

Dateline An extract that is just a date, which is most probably the dateline field of an article instead of its summary.

Shortness An extract that is trivially short.

Non-English An extract that isn't written in English.

3.2.2 Strapline detection heuristics

As mentioned in §3.1, one can distinguish straplines from summaries based on the common features that characterize each of them. As a way to automatically detect a range of straplines in the dataset, we present the following set of six rule-based heuristics:

Beginning with imperative speech One way to capture the reader's attention is to start a strapline with an imperative to read the article ("Check out ...").

Strapline characteristics: Clickbait, Little or redundant information.

Having high quotes coverage A common feature of a strapline is to quote a statement said by a person that is mentioned in the corresponding article or a quote that is related to the article's topic.

Strapline characteristics: Little or redundant information, Comment.

Using 1^{st} or 2^{nd} person pronouns Straplines may refer to the readers. This is done typically using 1^{st} and 2^{nd} person pronouns such as *you* and *we*.

Strapline characteristics: Clickbait, Joke, Informal.

Using question/exclamation marks Straplines are sometimes used to pose questions that stimulate the interest of the readers. On the contrary, summaries use objective sentences focusing on the main events of the articles, which makes it unlikely to find interrogative phrases in a summary.

Strapline characteristics: Little or redundant information, Joke.

Using a repeated extract Journalists tend to use the same strapline for an article that is being published on a regular basis (e.g.: a daily/weekly column or a message to the editor section). Consequently, an article with a non-unique extract indicates that the extract is a general statement, making it a strapline.

Strapline characteristics: Little or redundant information, General.

Using a clickbait Classifying an extract as a clickbait, as described in §4.2, can be employed to detect some of the extracts that are originally straplines.

Strapline characteristics: Clickbait.

Source	Summary	Strapline	Both	Neither	Paraphrase
NYT	87%	5%	3%	2%	3%
Time.com	48%	33%	6%	8%	5%
Combined	67.5%	19%	4.5%	5%	4%

Table 2: Distribution of extract annotations among labels on the annotated portion of the test set. Annotations were collected for 100 random samples from each source (NYT, and Time.com) resulting in a total of 200 annotated samples.

Round	Straplines		Summaries	
	Raw	κ	Raw	κ
1	0.70	0.36	0.72	0.37
2	0.82	0.55	0.80	0.49

Table 3: Inter-annotator agreement for strapline and summary annotations.

Source	Accuracy	Precision	Recall	Strapline%
NYT	90%	43%	75%	8%
Time.com	73%	68%	64%	39%

Table 4: Results of the rule-based strapline classification as a binary classification problem (Strapline/ Not Strapline).

Source		Noise	Strapline	Total
NYT	Training Set	899 (1.89%)	9,537 (20.07%)	47,529
	Test Set	101 (2.00%)	1,002 (19.86%)	5,045
Time.com	Training Set	937 (4.35%)	8,102 (37.61%)	21,541
	Test Set	108 (4.60%)	893 (38.03%)	2,348

Table 5: Number and % of noise and straplines our rule-based heuristics detected in NYT or Time.com data sections of Newsroom.

4 Experiments

4.1 Annotation

Two annotators[3] annotated 50 articles each from the NYT and time.com sections of the test set of Newsroom. We performed two rounds, resulting in a total of 200 articles with double annotation. In order to provide a single ground truth for the test set, the two annotators discussed their annotations and agreed on a single label for each article. For tuning the strapline detection method, we further annotated 50 articles each from the development sets of NYT and time.com sections.

Results Table 2 shows how often the annotators chose a particular label for the different news sources. Proper summaries are the largest class for both news sources, but Time.com has a considerably higher proportion of undesired straplines, and also a higher proportion of extracts that are both summaries as well as straplines.

In order to see how reliable the extracts can be annotated, we compute inter-annotator agreement between the two annotators. Table 3 shows the results for two annotation rounds. We compute the agreement by splitting our annotation into two binary labels, namely straplines vs. non-straplines, and summaries vs. non-summaries, excluding paraphrases. We report the proportion of labels that are the same for both annotators ("Raw" in the table), and Cohen's κ (Cohen, 1960), which accounts for agreement that is expected by chance. The results in Table 3 show that the agreement is high, but due to the class imbalance a sizable part of that high agreement might be due to chance (low κ value). However, the results show improvements in the consistency between the two annotators in the second round.

[3]The annotators are authors of this paper who were not involved in the development of the heuristics and the person responsible for the heuristics did not look at the annotations.

4.2 Strapline detection

Given the lack of annotated data for training a supervised strapline classification model, we implement a rule-based classifier by marking an extract as a strapline if any of the heuristics described in §3.2.2 apply to it. For the clickbait detector, we fine-tune the distilled BERT (Sanh et al., 2019) on the Webis-Clickbait-17 (Potthast et al., 2018) dataset and incorporate it into our strapline detector.

Results Table 4 shows the evaluation result of the strapline detector on the human annotated test set. We can observe that NYT test set is unbalanced where only 8 out of 100 samples are annotated as straplines, which also explains the difference between the accuracy and precision/recall. Time.com set is more balanced, and we can see that our model achieves a good performance with a precision of 68% and recall of 64%.

We apply the strapline detector on the training set to exclude the noisy samples and straplines. The result is shown in Table 5. We can observe that 20.07% samples of NYT and 37.61% of Time.com are classified as straplines, which shows that the strapline is an issue that cannot be ignored in the summarization dataset.

Training set		Original Test Set			Cleaned Test Set		
		R-1	R-2	R-L	R-1	R-2	R-L
NYT	original	13.57	3.03	11.60	12.25	1.09	10.16
	w/o straplines	**20.83**	**4.69**	**16.29**	**22.39**	**5.31**	**17.30**
Time.com	original	**15.96**	3.28	**13.39**	19.09	4.28	**15.83**
	w/o straplines	15.87	**3.34**	13.27	**19.12**	**4.32**	15.81
Combined	original	**19.16**	**4.89**	**15.58**	20.24	4.50	16.03
	w/o straplines	19.06	4.13	15.25	**21.29**	**4.94**	**16.82**

Table 6: ROUGE-1, ROUGE-2, and ROUGE-L scores for the abstractive summarizer (T5-base version) trained on the dataset with and without the straplines. The best results are in **bold**.

4.3 Summarization with cleaner data

We employ the most popular pre-trained sequence-to-sequence model, T5 (Raffel et al., 2019), as the basic summarizer in our experiments. We exclude the noisy samples and straplines by our proposed strapline detector (§4.2) from the NYT and Time.com dataset, forming a cleaner training set. We use T5-base and T5-large model in our experiments. We fine-tune them on the original and the cleaned dataset to see the influence of excluding noise and straplines. We use ROUGE (Lin, 2004a,b) to automatically evaluate the performance of the summarizers.

Results Table 6 shows the ROUGE-1, ROUGE-2, and ROUGE-L scores for the (T5-base) summarizer trained on the original training set and the cleaned training set[4]. We can observe that the impact of straplines on NYT is more significant than Time.com. For Time.com dataset, most ROUGE scores increase slightly by excluding the straplines. However, performance on NYT is greatly improved by up to 7 points. In part this is due to a repetition problem that we observe specifically on NYT: the model trained on the original data re-uses some summaries multiple times, with a single re-occurring sentence accounting for 10% of generated outputs whereas all summaries of the model trained on the cleaned data are unique. That is, the model seems to perpetuate the property of repeating extracts in the training data (see §3.2.2).

Case study For each news source, we manually compare the output of two T5-base models fine-tuned on the articles of the news source in the original dataset $M_{original}$ and the cleaned one M_{clean} in order to investigate the effect of excluding noise and straplines from Newsroom. Table 7 demonstrates the differences between the generated summaries by T5-base models that are fine-tuned on articles of each news source. The "Output of Original Model" $M_{original}$ column refers to the summaries generated by a model fine-tuned on the articles of Newsroom from the news source specified in the first column. On the other hand, the "Output of Cleaned Model" M_{clean} column refers to the summaries generated by a model fine-tuned on the articles of Newsroom from the news source after discarding the articles whose extracts are flagged as noisy or as straplines. We found two main improvements in the quality of the generated summaries: (i) M_{clean} tend to be more informative in compared to $M_{original}$ and (ii) M_{clean} do not exhibit as much undesired characteristics of straplines like: using a repeated summary, using a question mark, and using the 1st person pronouns, while $M_{original}$ tend to have such properties. The fact that these improvements do not have huge impact on the automatic evaluation metric (ROUGE) for Time.com implies that human evaluation is needed to accompany the automatic evaluation metrics in order to quantify such qualitative improvements.

5 Conclusion

We present methods for improving the data quality in scraped news summarization corpora, focusing on the New York Times and Time magazine sections of Newsroom (Grusky et al., 2018). We identify two main issues with the data quality that make Newsroom less appropriate as a summarization benchmark: (i) noise in the extraction process and (ii) presence of straplines in place of genuine summaries. After identifying common characteristics of straplines, we develop a set of effective heuristics for detecting straplines and noise.

Our work shows that when straplines and noisy data are excluded from the training data, the result-

[4]The corresponding scores for the T5-large summarizer are reported in Table 1 in the Appendix.

News source	Output of Original Model ($M_{original}$)	Output of Cleaned Model (M_{clean})
NYT	**A day in the life of** a Olympic athlete.	The Australian swimmer Mack Horton was booed by Chinese swimmers after his victory in the 200-meter freestyle, and Russian swimmer Irina Efimova was booed.
NYT	**A day in the life of** a Yankees fan.	The Yankees victory parade on Friday was a celebration of the teamâs success, but not everyone was there.
NYT	**A New York Times blog about** comic book publishing and design.	Kevin Conroys performances as Batman in the comic books, movies and television series stand out.
NYT	**New York Times reporters and editors are reporting from Washington, D.C.**	A New Hampshire biologist turned to film school to learn how to communicate scientific information.
NYT	**Reading, watching, discussing and blogging the day's local, national, and international news at The New York Times.**	The University of Illinois, Chicago, has a bright spot in its diversity.
NYT	**To the Editor:.**	Readers respond to an Op-Ed article about climate talks.
NYT	**To the Editor:.**	Readers responded to a recent editorial about the dangers of concealed carry.
Time.com	TIME 100 poll: **Who is the world's most influential leader?**	The Russian president has risen to second place in the TIME 100 poll, beating out world leaders like Pope Francis and Barack Obama
Time.com	California is cutting back on its water use, but **where is it going?**	California is cutting back on water usage by 25%, but the state isn't out of water
Time.com	A new survey shows that Millennials are becoming more entrepreneurial, but **we** need to do more to prepare them	A new survey finds that 82 percent of Millennials are interested in starting their own businesses
Time.com	A new report finds that more and more counties aren't affordable. Here's what **you** need to know	A new report finds that 9% of U.S. counties aren't affordable

Table 7: Example summaries selected from the outputs of the model fine-tuned on the original dataset and the cleaned dataset. Spans showing characteristics of straplines are **underlined and shown in bold text**.

ing summarizer produces better summaries based on comparison to reference texts. Although we found noise and straplines to be more prevalent in the Time magazine data, the impact of removing noise and straplines is bigger for the model trained on the NYT data, which avoids reusing the same summary multiple times. We plan to investigate this further in future work.

Because of our focus on two specific news sources in Newsroom, we suspect that our heuristics might not work quite as well on other news sources having different styles, or on other datasets that were collected differently.

Acknowledgments

This work was supported in part by the UKRI Centre for Doctoral Training in Natural Language Processing, funded by the UKRI (grant EP/S022481/1) and the University of Edinburgh, School of Informatics and School of Philosophy, Psychology & Language Sciences.

References

Rishi Bommasani and Claire Cardie. 2020. Intrinsic evaluation of summarization datasets. In *EMNLP*.

Jacob Cohen. 1960. A coefficient of agreement for nominal scales. *Educational and psychological measurement*, 20(1):37–46.

Martin Ester, Hans-Peter Kriegel, Jörg Sander, Xiaowei Xu, et al. 1996. A density-based algorithm for discovering clusters in large spatial databases with noise. In *kdd*, volume 96, pages 226–231.

Max Grusky, Mor Naaman, and Yoav Artzi. 2018. Newsroom: A dataset of 1.3 million summaries with diverse extractive strategies. In *Proceedings of the 2018 Conference of the North American Chapter of*

the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers), pages 708–719, New Orleans, Louisiana. Association for Computational Linguistics.

Matthew Honnibal and Ines Montani. 2017. spaCy 2: Natural language understanding with Bloom embeddings, convolutional neural networks and incremental parsing. To appear.

Taehee Jung, Dongyeop Kang, Lucas Mentch, and Eduard H. Hovy. 2019. Earlier isnt always better: Sub-aspect analysis on corpus and system biases in summarization. *ArXiv*, abs/1908.11723.

Ryuji Kano, Takumi Takahashi, Toru Nishino, Motoki Taniguchi, Tomoki Taniguchi, and Tomoko Ohkuma. 2021. Quantifying appropriateness of summarization data for curriculum learning. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 1395–1405, Online. Association for Computational Linguistics.

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

Chin-Yew Lin. 2004a. Looking for a few good metrics: Automatic summarization evaluation-how many samples are enough? In *NTCIR*.

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

Ilya Loshchilov and Frank Hutter. 2018. Fixing weight decay regularization in adam.

Ramesh Nallapati, Bowen Zhou, Cicero dos Santos, Çağlar Gulçehre, and Bing Xiang. 2016. Abstractive text summarization using sequence-to-sequence RNNs and beyond. In *Proceedings of The 20th SIGNLL Conference on Computational Natural Language Learning*, pages 280–290, Berlin, Germany. Association for Computational Linguistics.

Anjali Narayan-Chen, Prashant Jayannavar, and Julia Hockenmaier. 2019. Collaborative dialogue in Minecraft. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5405–5415, Florence, Italy. Association for Computational Linguistics.

Martin Potthast, Tim Gollub, Kristof Komlossy, Sebastian Schuster, Matti Wiegmann, Erika Patricia Garces Fernandez, Matthias Hagen, and Benno Stein. 2018. Crowdsourcing a large corpus of clickbait on twitter. In *Proceedings of the 27th international conference on computational linguistics*, pages 1498–1507.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2019. Exploring the limits of transfer learning with a unified text-to-text transformer. *arXiv preprint arXiv:1910.10683*.

Alexander M. Rush, Sumit Chopra, and Jason Weston. 2015. A neural attention model for abstractive sentence summarization. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 379–389, Lisbon, Portugal. Association for Computational Linguistics.

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

Nakatani Shuyo. 2010. Language detection library for java.

Karen Spärck Jones. 1999. Automatic summarising: factors and directions. In *Advances in automatic text summarisation*. MIT Press.

Priyam Tejaswin, Dhruv Naik, and Pengfei Liu. 2021. How well do you know your summarization datasets? In *FINDINGS*.

A Implementation details of noise filtration and strapline detection heuristics

Before applying the noise filtration and the strapline detection heuristics, Spacy's model (namely `en_core_web_sm`) (Honnibal and Montani, 2017) was used to tokenize the extracts, and determine the pos tags of the tokens.

A.1 Noise filtration

Web formatting syntax The following regular expressions `<[a-zA-Z0-9_]+[/]?>`, and `[a-z]+="` were used to determine the presence of HTML tags and key/value pairs as part of the extract. The first one looks for opening HTML tags in the form `<ALPHA_NUMERIC_SYBMBOL>`, and closing HTML tags in the form `<ALPHA_NUMERIC_SYBMBOL/>`. The second regular expression looks for alphabetic symbols followed by an equal sign and a double quotation.

Truncation An extract is considered to be truncated if it ends with a comma or ends with a word whose part of speech (pos) tag is a determiner, a coordinating conjunction, a subordinating conjunction, or an unknown pos tag.

Dateline Since dates might have different formats, a python package called *dateutil* [5] was used to parse the extract. An extract is considered as a dateline if the package manages to parse it according to any of the package's formats for dates.

Shortness Extracts having three or less tokens (after excluding punctuation marks) are considered to be trivially short and thus removed from the dataset.

Non-English On looking at the unique characters of the Newsroom dataset, we noticed that it contains characters from other scripts such as: Arabic, and Chinese. Consequently, a python package called *langdetect* [6] which is ported from one of Google's projects (Shuyo, 2010) was used in order to filter-out articles that aren't written in English. The article's text was used instead of the extract to detect the language, since the *langdetect* package has higher precision when supplied with longer spans of text (i.e. when given the whole article text instead of just the extract). This implies that we are assuming that the language of the article's body and its extract will be the same, and that having a non-English body is enough to discard the article-extract pair from the dataset.

A.2 Strapline detection

Beginning with imperative speech If the pos tag of the first token in the extract is VB (base form of verb), then the extract is considered to be beginning with an imperative.

Having high quotes coverage A simple pattern matching function is used to compute the percentage of the tokens found between quotes in the extract. An extract is considered as a strapline if its quotes coverage is higher than a preset threshold (a hyperparameter set to 0.35 based on manual investigations of the dataset).

Using 1^{st} or 2^{nd} person pronouns If any of the extracts' tokens is part of the following list (i, me, mine, myself, we, our, ours, ourselves, you, your, yours, yourself, yourselves), then it's said to use a 1^{st} or 2^{nd} person pronouns.

Using question/exclamation marks The presence of a question or an exclamation mark is used to simplify the detection of interrogative/ exclamation phrases.

Using a repeated extract If an extract is repeated more than once in the training dataset then it's discarded. Using a clustering method such as Density-Based Spatial Clustering of Applications with Noise (DBSCAN) (Ester et al., 1996) on top of sparse term frequency vectors representing the extracts achieves better performance at the expense of running time. Therefore, we opted to use the simple method of having exact matches as a method to detect repeated extracts.

B Hyperparameters in the experiments

Clickbait Detector We fine-tune distilled BERT using AdamW optimizer (Loshchilov and Hutter, 2018), the early stopping mechanism with patience of 5, a batch size of 128, and a learning rate of 10^{-4}. The max input length is set to 512.

T5-based Summarizer The max length of input and output are set to 512 and 128, respectively. We fine-tune T5 using AdamW optimizer (Loshchilov and Hutter, 2018), the early stopping mechanism with patience of 5, a batch size of 32, and a learning rate of 10^{-4}.

C Results of fine-tuning T5-large

Looking at the ROUGE scores in Table 1, one can notice that similar trends are achieved on fine-tuning a T5-large summarizer to these found on fine-tuning a T5-base summarizer (as discussed in the main paper). While T5-large achieves higher absolute ROUGE scores, the effect of removing noise, and straplines from the training corpus is nearly the same for both the T5-base, and the T5-large models, which demonstrates that more attention needs to be given to the quality of the dataset rather than using larger models.

D Distribution of Heuristics

Table 2 shows the distribution within the NYT and Time.com datasets, including both noisy samples and straplines. Note that there might be overlap between different heuristics.

[5] https://dateutil.readthedocs.io/en/stable/

[6] https://pypi.org/project/langdetect/

Training Set		Original Test Set			Cleaned Test Set		
		R-1	R-2	R-L	R-1	R-2	R-L
NYT	original	16.62	4.35	13.63	15.87	2.59	12.56
	w/o straplines	**21.80**	**5.30**	**17.19**	**23.43**	**6.03**	**18.34**
Time.com	original	**16.47**	**3.44**	**13.64**	**19.36**	4.32	15.82
	w/o straplines	16.07	3.38	13.43	19.28	**4.46**	**15.96**
Combined	original	**20.19**	**5.50**	**16.41**	21.54	5.25	17.05
	w/o straplines	19.61	4.60	15.79	**22.07**	**5.55**	**17.60**

Table 1: ROUGE-1, ROUGE-2, and ROUGE-L scores for the abstractive summarizer (T5-large version) trained on the dataset with and without the straplines. The best results are in bold.

Heuristic		NYT		Time.com	
		Training Set	Test Set	Training Set	Test Set
Noise	too_short	**1.42%**	**1.55%**	**3.61%**	**3.92%**
	is_a_date	0%	0%	0.32%	0.43%
	has_HTML	0.09%	0.06%	0.55%	0.55%
	strange_ending	0.09%	0.12%	0.26%	0.21%
	is_non_english	0.31%	0.34%	0.01%	0%
Strapline	mostly_quotes	0.03%	0.06%	0.15%	0.22%
	has_1st_or_2nd_person_pronoun	**6.80%**	**7.54%**	14.11%	14.60%
	has_question_exclamation_marks	5.69%	6.05%	6.08%	5.67%
	imperative_speech	1.07%	1.01%	4.12%	4.68%
	is_repeated	5.78%	4.43%	0%	0%
	is_clickbait	6.34%	6.53%	**29.03%**	**28.75%**

Table 2: The distribution of the heuristics (both noises and straplines) within the datasets.