Got Headlines?
Abstractive Title Generation for News Articles

Aniruddh Rao and **Manav Bhatia**
University of Michigan, Ann Arbor
{anrao,manavrb}@umich.edu

Abstract

Title generation is a challenging task that involves creating short, interesting titles for news articles that capture the important information in the article while still leaving enough information hidden to make the full article worth reading. In this paper, we demonstrate the use of the underexplored ALL THE NEWS 2.0 dataset, with 2.7 million articles from 27 American publications, to train and evaluate models for title generation. We finetune T5, Pegasus, and Pegasus-X models and evaluate their performance using the ROUGE metric and human evaluation. We also analyze the relationship between temporal drift in article context and the quality of generated titles. Our results suggest that the variance in style and targets of training data effects title generation quality and should be considered. Our results also indicate that continual training on a new year's article prevents degradation of title generation quality at best.

1 Introduction

Generating informative and interesting titles is a challenging component of the pipeline for publishing a textual article and involves human creativity. On the surface the problem of title generation seems to be reducible to the more general and more well-researched field of text summarization. Contrary to that intuition, title generation is a more nuanced sub-task of the problem. This nuance comes from the fact that titles are meant to be short for consumers skimming a newspage and also to accomodate users on smaller devices. Additionally, with advertising being a huge part of the earnings for a publication house, click-through rates for articles based on title is a priority. This then requires titles to be interesting enough to make the article worth reading, which rewards not giving *all* of the important information in an article.

As such, title generation becomes a balancing act between capturing all important information in an article while hiding away enough information to make the full article worth reading. Therefore, we aim to create and analyze a model to abstractively generate titles for textual news articles from various publications. As such we make the following contributions in this paper:

- Demonstrated the use training on ALL THE NEWS 2.0, a relatively underexplored dataset with 2.1 million articles from 27 different American publications;
- Finetuned T5, Pegasus and Pegasus-X models to generate titles for news articles and evaluate generation via the ROUGE metric (Lin, 2004) and human-evaluation;
- Provided discussion on Pegasus-X versus Pegasus on short abstractive text summarization in context of a diverse dataset, using the ROUGE evaluation framework and human evaluation;
- Analyzed the relationship betwween temporal drift in article context and the quality of titles generated by training the model on data from one year and evaluating on the next, for data from 2016 to 2020.

2 Previous Work

There has been some previous work done in this area. (Xu et al., 2019) uses reinforcement learning to generate "sensational headlines" or click bait style headers.

(Aditi et al., 2021) was much more closely aligned to what we are trying to accomplish using Vanilla RNN and versions of LSTM to train. However their dataset is much less extensive and there is a lacking in discussion of evaluation metrics. We would look to improve on both of these missing points as well as make more concrete conclusions on how generic title generation can be accomplished.

(Gu et al., 2020) is a great previous work that provided us a building block for us to spring off from. It does many of the same things we intended to do but stops short from anything further than generating a title. The dataset used was also much smaller and more limited than ALL THE NEWS 2.0, which will give us the opportunity to test more variables.

(Lopyrev, 2015) does a similar analysis to ours but used the Stanford Linguistics Gigaword dataset which is a limited access dataset and trained a single direction RNN. They notably found that the model did not perform on general text, and we believe our more modern approach aims to fix this artifact.

Finally, the area of abstractive text summarization using large language models is a well-researched field. As such, (Zhang et al., 2019), (Raffel et al., 2019a), (Phang et al., 2022) provide the T5, Pegasus and Pegasus-X models respectively, for the task of abstractive text summarization. That said these works have not focused on title generation in specific and largely use the CNN/DAILYMAIL or NEWSROOM datasets for training and testing, which are generally smaller and less diverse than ALL THE NEWS 2.0. Further, we note that (Phang et al., 2022) mention that while they noticed Pegasus outperform Pegasus-X on shortform summarization tasks like ours, they hoped to compare the two in context of a more diverse dataset and expected Pegasus-X to outperform Pegasus. We hope to answer this question given that we do in fact have a more diverse dataset for training and evaluation.

3 Dataset

ALL THE NEWS 2.0 (Andrew, 2020) is the dataset we used for training and evaluating and it provides approximately 2.7 million articles from 27 different American publications. The features from this dataset that we use are the article text, the publication name, the year of publication and the published title. We note that the dataset is largely balanced in terms of year of publication as seen in Figure 1 but is imbalanced in terms of source publication.

We note that the largest clusters for the publications are Reuters, The New York Time, CNBC and The Hill. A deeper look into the value counts can be found in the figures in Appendix A.

For our first set of experiments, we picked out large clusters of publications and use these articles as our first working dataset. For this, the publication clusters we chose were Reuters, CNN and The New York Times. This justified by the fact that these publications are the most 'general' in terms of topics in addition to having a lot of articles in the overall dataset. We considered including the third largest cluster of CNBC articles but chose not to since CNBC focuses on business news and we wanted to focus on publications that focused on a wide coverage of topics for the first experiment. Given this, we split this group of general publications into training, validation and testing sets with 80-10-10 splits. Finally, we also take 10% of each of the left out publication's articles and use those or evaluation as well.

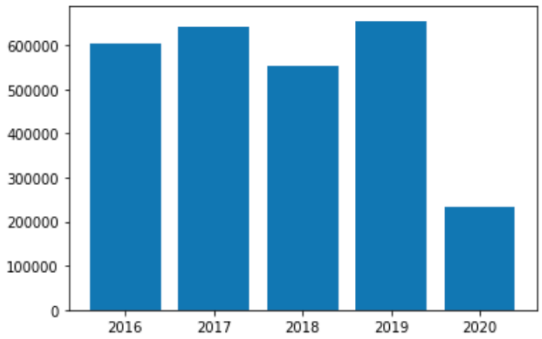


Figure 1: Value Counts for for year of publication of an article in ALL THE NEWS 2.0

For the next set of experiments, we took all the articles from every publication and divided them by year. We then split each year's data into training, validation and testing sets with a 80-10-10 split. We then trained and evaluated our models **continually** on the data for each year to analyze the effect of temporal drift in context of articles and how well the model can perform on data from unseen years.

4 Methodology

The subsections below cover the pipelines, with justifications for the same, for our 2 experiments. We also provide a description for the Human Evaluation we did for all of our best models from each experiment in the last subsection.

4.1 Generalization Experiments

With our first set of experiments we had two main questions we wanted to answer:

- Does title generation generalize well to other styles or articles targetted to a specific area e.g. Business or Politics?

- Can the Pegasus model outperform Pegasus-X for short abstractive summarization tasks like this, when given (more) diverse data than CNN/DAILYMAIL?

As mentioned in the datasets section, we first make a working dataset with articles from solely CNN, The New York Times and Reuters. These were picked since they form large clusters and are publications that cover a wide range of topics rather than targetting areas like politics.

We then picked 3 models to train on summarization, which were the T5 (Raffel et al., 2019a), Pegasus (Zhang et al., 2019) and Pegasus-X (Phang et al., 2022) models. We picked T5 to get a good baseline for our dataset and used the Pegasus models since they are known to be SOTA for the abstractive summarization tasks. More importantly, (Phang et al., 2022) demonstrate that Pegasus-X can summarize texts of any length using global attention mechanisms, eliminating the need for truncation like transformer models usually require. They then found that on shorter summarization tasks, like for CNN/DAILYMAIL headlines generation (a pure summary, **not** a title), original Pegasus outperforms the newer Pegasus-X model. The reason noted here was that with a more diverse dataset Pegasus-X might still outperform Pegasus for short summarization tasks like ours.

As such, we first note that we have a diverse dataset in terms of publication year and publication source, with this specific experiment's working dataset having 3 publications (more importantly, for the training). This with the fact that we have 24 very different out of distribution publication sources for articles to evaluate the model on, we hope to provide a better analysis for their comparison of Pegasus to Pegasus X on short summarization tasks.

Considering all of this, we utilized the data prepared as discussed above, pass it through each model's tokenizer provided by HuggingFace (Wolf et al., 2020), train the model on the training split for this experiment and finally evaluate the model using the ROUGE metric alongside human evaluation. We note the configurations for each of the models trained in Appendix B. We note in specific that all training was done using DeepSpeed ZeRO stage 2 (Raffel et al., 2019b) in order to speed up our multi-gpu training and evaluation process on the HPC clusters used.

For the evaluation we first used the ROUGE metric (Lin, 2004) to evaluate the models. In particular, given ROUGE can be calculated automatically, we provide ROUGE results for the testing split for CNN/NYT/Reuters split. We also provide the same for model evaluation done on the other articles in the dataset from the 24 unseen publication, many of which target specific areas of coverage. As noted in Section 3, we do this out of distribution testing on 10% of the articles from each of the 24 sources.

Finally, this process gives us models trained on our chosen cluster of publications and evaluated on all testing articles from out of distribution articles publications. We term the respective models as **T5 Ally**, **Pegasus Ally** and **Pegasus-X Ally**.

4.2 Year-By-Year Training Experiments

Given that our data was extremely balanced by year, as seen in Appendix A, we wanted to answer the following question:

- In a real-world scenario where the model is trained on data from the previous year, does the model benefit from training on the data from the new year?

We define the term **temporal drift** as change in the topics and objects spoken of in a news article as time goes on. This is an extremely important factor to consider when training models in our opinion since the context of news changes very quickly given the nature of the field. As such, we wish to see how well the model does on testing data from the current year it has been trained on (including previous year data) and the testing data from the next year it has not trained on. Then, training the model on the training data from next year and repeating the process tells us how well the model can title articles from the future, which it has no definite information on.

We picked the Pegasus X model for this process for two reasons. First, we still wanted to get further results on how well Pegasus X can do when given diverse data and two, it is valuable to require no truncation for news articles (which we found is required for our data when we use pegasus) given that information is highly valued and people's needs from a news article changes with time. As an example, a consumer might want longer articles with more details from business news sources when the general economy might be in recession.

Next, we divide our articles into folds based on the year of publication. This gives us 5 splits from 2016 to 2020, all of which are, once again,

fairly balanced. We then split these folds using a $80-10-10$ split once again. we also note that we do not train on articles from the year 2020 and only perform evaluation on it's testing split.

We use the same training and evaluation procedure as the last experiment to train on one year's training split, evaluate on that year's testing split and evaluate on the next year's testing split. We then repeat this process for this **trained model** this for the data from the next year and so on.

Finally, we do this evaluation using ROUGE and provide these results in the next section and generate a model that has been trained on all the training splits for the publications years of upto 2019. Given that this model covers most of our dataset, we include this in our final human evaluation and call this model **Pegasus-X 2019**.

4.3 Human Evaluation

While we use ROUGE for evaluating our two experiments above, ROUGE is known to not be entirely representative of actual human evaluation when it comes to summarization tasks and does not account for factuality of the titles generated. As such, while we decided to do most of the initial evaluation using this metric we also added human evaluation for all the notable models in each experiment at the end of our research. While we would have liked to entirely use human evaluatio, we chose to leave that for future work.

We took 4 articles from CNN, 5 articles from CNBC and 5 articles from the Onion. The CNN testing tests how well a model does on the general styled news articles, which it is familiar with due to the structure of our training. The CNBC articles then test how well a model does on articles that are more business focused and therefore is titling for a style more relevant to readers wanting business related information. Finally, we included the Onion (a website that parodies mainstream media) to see if our model can still title articles that are fake and entirely satirical in nature. We find this even more interesting considering that satire might value worse titles in general, making it hard to hypothesize how well our model *should* do on these articles.

For our human evaluation process we had 3 seperate testers (2 of which were authors of this paper) to evaluate the title generation power of Pegasus Ally, Pegasus-X Ally and Pegasus-X 2019. We chose not to include T5 in this experiment given that it was just a baseline for the ROUGE based evaluation. The key points the evaluators were asked to look for were readability, accuracy and if the title sounds interesting.

Each evaluator was assigned articles from one publication source. They were then given titles from our models and the original title, without telling them which title is the original or from a specific model. We then asked them to rank the titles before they read the article and after they read the article. The first phase captures how well the title reads or piques their interest, and the second captures the actual accuracy of the title in addition to the readability.

Finally, we present the exact details for this evaluation in Appendix D and the average rank for each model (and original titles) in Section 5.

5 Results

5.1 Generalization Experiments

First note that all mentions of ROUGE in this experiment refers to ROUGE-L.

Now, given Figure 2, out first notable finding is that Pegasus Ally seems to outperforming Pegasus X Ally in every category of publications, both in and out of distribution. This therefore tells us that contrary to the hypothesis in (Phang et al., 2022), Pegasus-X Ally does not outperform Pegasus when trained and evaluated on a more diverse dataset than CNN/DAILYMAIL. This is a notable finding since this implies that while Pegasus-X Ally achieves state of the art results in longform articles, shortform summarization is likely a different enough problem for the model to not be state of the art for it as well. This open a very interesting avenue for research on improving global attention mechanisms to perform equally well or better on shortform abstractive text summarization tasks. We provide another datapoint for this research in the year-by-year experiment, where we train Pegasus-X on an even more diverse dataset and then follow that by including that trained model in our human evaluation.

Next, using Figure 2 and Table 1, it seems that there are 2 different ways to evaluate generalization. The first way is to consider only the average differences in Table 1 without the results in Figure 2. In this sense, T5 Ally seems to have the lowest degradation in scores for out of distribution publication's articles, followed by Pegasus X Ally and finally Pegasus Ally. That said, we still (in a

ROUGE in−ROUGE out	T5 Ally	Pegasus Ally	Pegasus-X Ally
CNBC	10.3411	23.9671	20.9878
Politico	12.8747	18.1835	15.5905
TechCrunch	16.2547	23.0577	20.5582
Average	13.15683333	21.7361	19.0455

Table 1: Difference in ROUGE-L score for in distribution publication articles subtracted by ROUGE-L score for out-of-distribution publication articles, for each Ally model.

world where ROUGE is completely accurate) want the ROUGE to be high and not just minimize the ROUGE difference between the in distribtion and out of distribution publication titles. As such, via Figure 2, Pegasus achieves the best raw ROUGE scores for all the out of distribution publications as well as in-distribution publications. Further, in this regards T5 Ally does the worst.

That said, there is an exception for CNBC articles where T5 Ally outperforms all the models even in raw ROUGE scores. Our guess here is that T5 Ally performs less abstractively than the other models and the added extractiveness likely does well for a business or markets focused publication like CNBC that uses a lot of numbers and terse terms in its titles.

Given all of this, the takeaway here is that given that we value the titles to be of better quality in general rather than having less quality difference between general vs in-distribution articles, T5 Ally is the worst generalizer (except for CNBC) and Pegasus is the best. This trend also holds for the in distribution articles. We note that it might be worth testing more configurations for T5 Ally and more importantly, Pegasus X to find if they can achieve better raw ROUGE scores all around. If so, then they might be able to outperform Pegasus in generalization given that both their degradation on out of distribution publications is lower. That said, for now we use these results to eliminate T5 Ally from the human evaluation experiment pipeline.

5.2 Year-By-Year Training

The results for each of the years are illustrated in Figure 3. As mentioned in the previous section, we start train the model on all articles from the year 2016 and then evaluate on test articles from the previous (not applicable for 2016), current and next years. We then repeat this process by taking the last trained model and training it on the next year. As such, the figures list the ROUGE performance for each year on it's 3 evaluation years.

The first finding we note is that the models generally do worse on testing data from the next year than their current year very consistently. It is also worth noting that this difference is pretty inconsistent for the 4 years considered but is the worst for the model trained on 2019 data, where the ROUGE-L performance gap between 2019 and 2020 evaluation is 6.7144.

We then note that as you train continually on years, the model seems to forget the titling style or content for the previous years. This is clear given that the ROUGE-L score degrades by 1.0425 on average for the evaluations on the previous year after training on the next year. This makes sense since large transformer based language models are known to suffer from the problem of catastrophic forgetting, and trying to minimize this would likely be a good consideration for future work.

Next, we notice that models generally do better on the next unseen year after being trained on the next year's data using the next year's training data. The ROUGE-L improves by 2.958 points on average for evaluations on the test set before and after training for the new year.

These results motivate the question of how much this additional training on the new years (since you get new data year-by-year if you were a publication) matters. This is an even bigger consideration given that the model's performance on the next unseen year is very consistently worse even as you accumulate more examples in your training history. When considering the cost and time of training on the new year's data and then evaluating on the 3 different testing sets, our conclusion is that it likely is **worth training continually on data from your last year of articles** to at least **minimize your performance degradation**. That said, in case a publisher is limited on training and evaluation resources it would likely not result in large quality degradation if they chose not to keep training their model, based on our findings.

5.3 Human Evaluation

The human evaluation yielded some interesting results when we compared articles from the Onion, a satirical publication, CNBC, a more genre focused publication, and CNN, a more generic publication. We presented the testers with four titles for each article in a random order and asked them to rank them from best to worst. Then after reading the article the testers were given one more chance to adjust their rankings after knowing the contents of

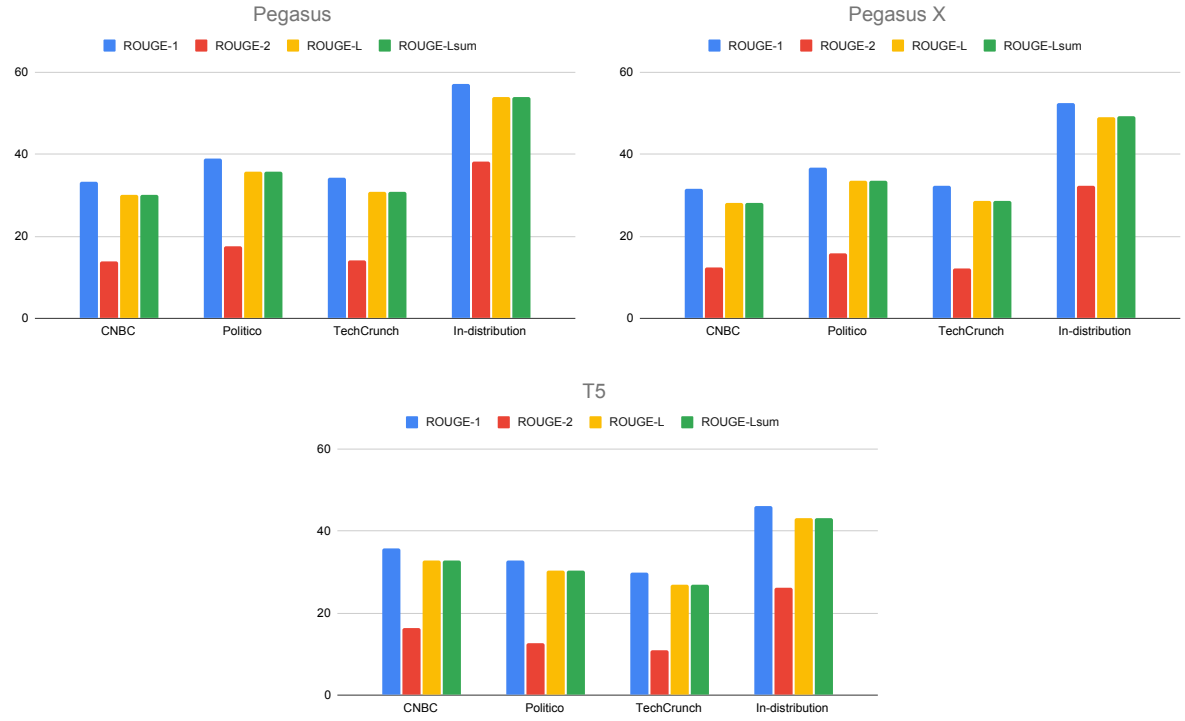


Figure 2: ROUGE scores for evaluations done on CNBC, Politico, TechCrunch and testing articles from Reuters-NYT-CNN fold of articles for T5 Ally (bottom), Pegasus-X Ally (top-right) and Pegasus Ally (top-left) models.

Figure 3: ROUGE scores for each evaluation done for models trained till 2016 (top-left), 2017 (top-right), 2018 (bottom-left) and 2019 (bottom-right)

the article. We noted down what rank each title received ($\{1, 2, 3, 4\}$ with 1 being the best and 4 being the worst) in each stage and then averaged them to get the average rank of the model's title. Below is an example of 4 generated titles for an article about Carvana shares from CNBC:

- **True Title**: Carvana shares tank as bankruptcy concerns grow for used car retailer
- **Pegasus**: Carvana Shares Plunge After Major Creditors Sign Binding Deal
- **Pegasus X**: Carvana's biggest creditors sign deal to help with debt restructuring
- **Pegasus-X 2019**: Carvana shares plunge after company's largest creditors sign deal

Models with a lower average number are the better models (1 is best and 4 is worst). We can see in Figure 4 the average ranking for each model's title. On the left is the averages without the data from the Onion included and on the right is the data from all three publications. The reason for this split is to see if the satirical style of the Onion articles and title may have wildly thrown off the results. Since these titles are meant to be over the top, it could be confusing for the tester to denote between a legitimate title and a non legitimate one.

The full numerical results can be found in Appendix D. As we can see from the figure, these results are very interesting. As a refresher, Pegasus Ally and Pegasus-X Ally were trained on all years on the three big **generic** publications: The New York Times, CNN, and Reuters. These publications don't cover one specific genre but a wide variety of topics and are major respected sources across the global mainstream media.

Pegasus-X 2019 was trained on **all publications** in the dataset on every year except 2019. Here, we can directly see if training on more information of more variety helped generate more realistic titles or not. We can see in Figure 4 that firstly, the actual article titles performed worse than both Pegasus Ally and Pegasus X Ally and only marginally better than Pegasus-X 2019. This inital result indicates that we were successfully able to train a model to generate titles accurately enough to convince humans that they were the actual publication titles for the article. There was only one instance in our testing when a tester ranked the true title in first place after reading the article and adjusting the rankings. The other surpising component here is that models trained on the big three generic publications significantly outperformed Pegasus-X 2019, which was trained on all of the publications in our dataset.

This leads us to conclude that training on more data doesn't necessarily produce better results. Lots of other publications are geared towards a specific genre and overloading the model with too many themes and styles produces a negative result. We find that the model is not able to accurately emulate titles. This leads us to conclude that training on a generic source with a large sample size is more effective and accurate in generating generic titles and text.

We also observe that the data from the Onion did not significantly impact the results and only caused a small gap between Pegasus Ally and Pegasus-X Ally, showing that the former outperforms the latter in satirical text generation and summary. Overall, Pegasus Ally was the winner followed by Pegasus-X Ally, the true title and Pegasus-X 2019 in that order. Training data specificity matters for the application of the model and more data does not produce better results.

6 Conclusion

In this paper we present models to perform the abstractive summarization task of title generation for news articles. We present a finetuning pipeline using 3 different models for an large, diverse and underexplored dataset of news articles from 27 different American publications. We train 3 different models on articles from publications with general coverage of most topics. Further, we provide an analysis on the relationship between temporal drift in article content and quality of titles in terms of ROUGE. We found from our results that 1) The generality and style of the training articles effects the quality of titles generated 2) Training on a large amount articles from a new year prevents degradation in title generation quality but little to no improvements 3) Pegasus-X with a diverse dataset for training and evaluation, does not outperform Pegasus on shortform abstractive text summarization, contrary to the hypothesis by (Phang et al., 2022).

7 Codebase

All code for this project can be found in this GitHub repository.

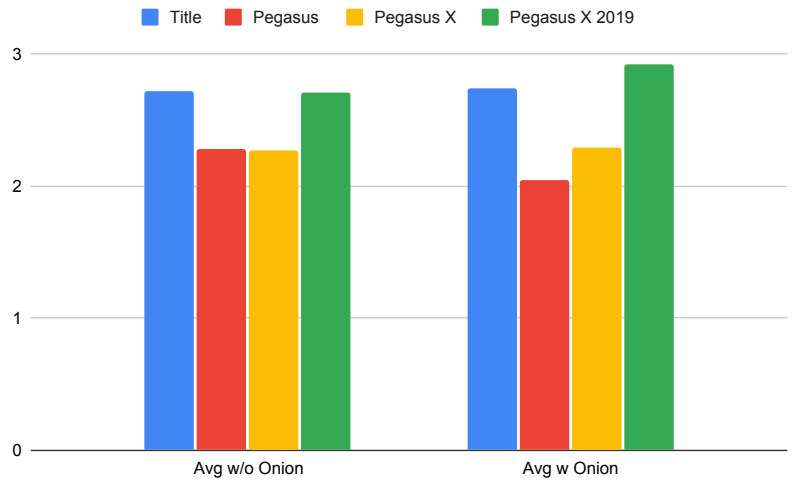


Figure 4: Human Evaluation Results for Pegasus Ally, Pegasus-X Ally, Pegasus-X 2019 and the actual title, where a *lower* average rank indicates a better title

References

Singh Aditi, Agarwal Ishita, Khan Kaamraan, and Desai Vedant. 2021. Title generation using NLP.

Thompson Andrew. 2020. All the news 2.0 — 2.7 million news articles and essays from 27 american publications.

Xiaotao Gu, Yuning Mao, Jiawei Han, Jialu Liu, Hongkun Yu, You Wu, Cong Yu, Daniel Finnie, Jiaqi Zhai, and Nicholas Zukoski. 2020. Generating representative headlines for news stories.

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Annual Meeting of the Association for Computational Linguistics*.

Konstantin Lopyrev. 2015. Generating news headlines with recurrent neural networks.

Jason Phang, Yao Zhao, and Peter J. Liu. 2022. Investigating efficiently extending transformers for long input summarization.

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

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

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 38–45, Online. Association for Computational Linguistics.

Peng Xu, Chien-Sheng Wu, Andrea Madotto, and Pascale Fung. 2019. Clickbait? sensational headline generation with auto-tuned reinforcement learning.

Jingqing Zhang, Yao Zhao, Mohammad Saleh, and Peter J. Liu. 2019. Pegasus: Pre-training with extracted gap-sentences for abstractive summarization.

BATCH_SIZE	NUM_EPOCHS	learning_rate	weight_decay	gradient_accumulation
16	2	8e-4	0.01	4

Table 2: Training Configurations

A All The News 2.0

Provided in Figure 5 is a visulization of the value counts of each publication name in ALL THE NEWS 2.0 (Andrew, 2020) and Figure 1 from earlier in the paper illustrates the balance of the same dataset by year of publication.

We also note that we received explicit permission from the authors of this dataset for use in our research.

B Model Configurations

In Table 2 we provide the configurations used for training our models, which were largely the same as the defaults recommended by the authors for summarization tasks. Note that we used bf16 mixed precision on the spgpu partition for all of our training and evaluation. Further we used DeepSpeed ZeRO-2 (Raffel et al., 2019b) for all our training. Also note that for all evaluations generations we used a max title length of 100 along with default beam search size.

C Generalization Experiments

Tables 3, 4 and 5 provide the tables and specific ROUGE score values for the graphs in Figure 2 on CNBC, Politico and TechCruch for the models T5 Ally, Pegasus Ally and Pegasus X Ally respectively.

D Human Evaluation

Figure 6 gives an example of the human evaluation process for a human evaluator. The evaluators were given the titles with the model names hidden and asked to enter the index of the best title in one of the indices 1-4 on their grading sheet to rate that title with a rating of $1-4$ (1 best, 4 worst) respectively.

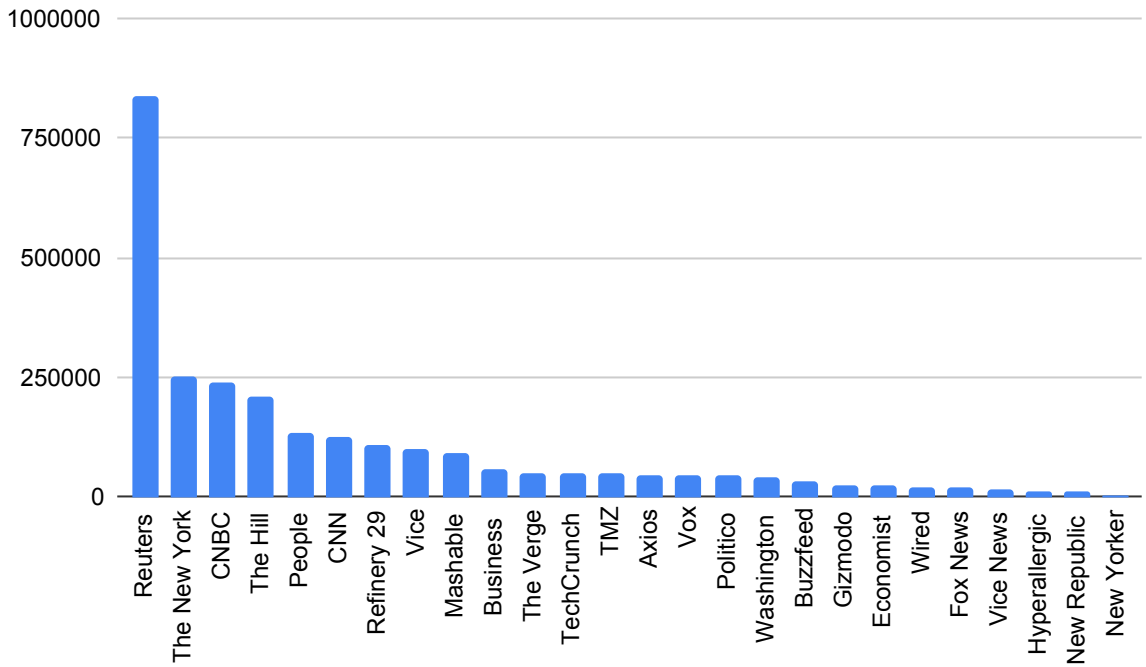


Figure 5: Value counts for the Publication Names in ALL THE NEWS 2.0

	ROUGE-1	ROUGE-2	ROUGE-L	ROUGE-Lsum
CNBC	35.6817	16.2941	32.8344	32.8643
Politico	32.8157	12.5475	30.3008	30.3362
TechCrunch	29.728	10.8447	26.9208	26.9367
In-distribution	46.151	26.2163	43.1755	43.2193

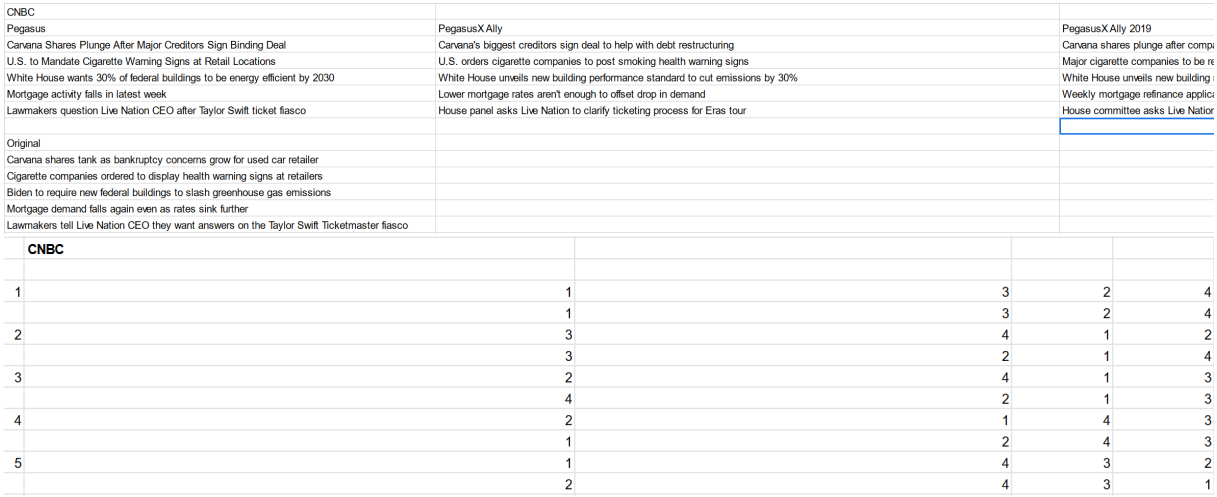
Table 3: T5 Ally Generalization ROUGE Score Results

	ROUGE-1	ROUGE-2	ROUGE-L	ROUGE-Lsum
CNBC	33.3888	13.7834	29.9806	30.0625
Politico	38.9653	17.6161	35.7642	35.8201
TechCrunch	34.278	14.2049	30.89	30.9233
In-distribution	57.1112	38.1159	53.9477	54.0248

Table 4: Pegasus Ally Generalization ROUGE Score Results

	ROUGE-1	ROUGE-2	ROUGE-L	ROUGE-Lsum
CNBC	31.578	12.4007	28.0806	28.1688
Politico	36.8122	15.8674	33.4779	33.5534
TechCrunch	32.3612	12.1696	28.5102	28.581
In-distribution	52.5654	32.2633	49.0684	49.1513

Table 5: Pegasus-X Ally Generalization ROUGE Score Results



CNBC		
Pegasus	PegasusX Ally	PegasusX Ally 2019
Carvana Shares Plunge After Major Creditors Sign Binding Deal	Carvana's biggest creditors sign deal to help with debt restructuring	Carvana shares plunge after compa
U.S. to Mandate Cigarette Warning Signs at Retail Locations	U.S. orders cigarette companies to post smoking health warning signs	Major cigarette companies to be re
White House wants 30% of federal buildings to be energy efficient by 2030	White House unveils new building performance standard to cut emissions by 30%	White House unveils new building
Mortgage activity falls in latest week	Lower mortgage rates aren't enough to offset drop in demand	Weekly mortgage refinance applica
Lawmakers question Live Nation CEO after Taylor Swift ticket fiasco	House panel asks Live Nation to clarify ticketing process for Eras tour	House committee asks Live Nation
Original		
Carvana shares tank as bankruptcy concerns grow for used car retailer		
Cigarette companies ordered to display health warning signs at retailers		
Biden to require new federal buildings to slash greenhouse gas emissions		
Mortgage demand falls again even as rates sink further		
Lawmakers tell Live Nation CEO they want answers on the Taylor Swift Ticketmaster fiasco		

	CNBC			
1	1	3	2	4
	1	3	2	4
2	3	4	1	2
	3	2	1	4
3	2	4	1	3
	4	2	1	3
4	2	1	4	3
	1	2	4	3
5	1	4	3	2
	2	4	3	1

Figure 6: Figure on the top indicates the titles given to the evaluator, albeit with the model names hidden and the figure on the bottom shows how an evaluator grades the 5 titles they are given