TLDR: Extreme Summarization of Scientific Documents

Isabel Cachola[†] **Kyle Lo**[†] **Arman Cohan**[†] **Daniel S. Weld**[†‡]

[†]Allen Institute for Artificial Intelligence

[‡]Paul G. Allen School of Computer Science & Engineering, University of Washington

{isabelc,kylel,armanc,danw}@allenai.org

Abstract

We introduce TLDR generation for scientific papers, a new automatic summarization task with high source compression, requiring expert background knowledge and complex language understanding. To facilitate research on this task, we introduce SCITLDR, a dataset of 3.9K TLDRs. Furthermore, we introduce a novel annotation protocol for scalably curating additional gold summaries by rewriting peer review comments. We use this protocol to augment our test set, yielding multiple gold TLDRs for evaluation, which is unlike most recent summarization datasets that assume only one valid gold summary. We present a training strategy for adapting pretrained language models that exploits similarities between TLDR generation and the related task of title generation, which outperforms strong extractive and abstractive summarization baselines.[1]

1 Introduction

We introduce the task of TLDR generation of scientific papers.[2]

An alternative to abstracts, TLDRs of scientific papers leave out nonessential background or methodological details and capture the key important aspects of the paper, such as its main contributions. This is similar to the tasks of extreme summarization (Narayan et al., 2018) and title generation (Vasilyev et al., 2019), which seek to produce single phrase or sentence summaries of news articles.

Writing a TLDR of a scientific paper requires expert background knowledge and complex domain-specific language understanding to identify the salient aspects of the paper, while maintaining faithfulness to the source and correctness of the written summary. An example of a TLDR is shown in Figure 1.

Abstract While many approaches to make neural networks more fathomable have been proposed, they are restricted to interrogating the network with input data. Measures for characterizing and monitoring structural properties, however, have not been developed. In this work, we propose neural persistence, a complexity measure for neural network architectures based on topological data analysis on weighted stratified graphs. [...]

Intro [...] In this work, we present the following contributions: We introduce neural persistence, a novel measure for characterizing the structural complexity of neural networks that can be efficiently computed. We prove its theoretical properties, such as upper and lower bounds, thereby arriving at a normalization for comparing neural networks of varying sizes. [...]

Conclusion [...] However, this did not yield an early stopping measure because it was never triggered, thereby suggesting that neural persistence captures salient information that would otherwise be hidden among all the weights of a network [...]

TLDR We develop a new topological complexity measure for deep neural networks and demonstrate that it captures their salient properties.

Figure 1: A TLDR is an extreme summary of a scientific paper. Above is an example of a TLDR and its corresponding paper. TLDRs could involve paraphrasing and abstraction from multiple sentences from the paper to convey the main topic or contribution of a given paper in a concise way.

To support this proposed task, we introduce SCITLDR, a dataset of 3,935 TLDRs in the scientific domain. SCITLDR is built from a combination of TLDRs written by human experts and author-written TLDRs of computer science papers from OpenReview.[3] Most summarization datasets provide a single gold summary for a given document, despite early work in summarization evaluation identifying variety in human-generated summaries (Zechner, 1996; Harman and Over, 2004). We consider this overly simplistic, especially for a high-compression task like TLDR generation. To address

[1] https://github.com/allenai/scitldr

[2] TLDR is an acronym that stands for "too long; didn't read," which is often used in online informal discussion (e.g., Twitter or Reddit) about scientific papers. For visual clarity, we omit the semi-colon.

[3] https://openreview.net/

this limitation, SciTLDR includes multiple gold TLDRs for each paper in the test set, where one TLDR is written by the author while the rest are human-written TLDRs that are obtained from peer review comments. In the latter type of TLDRs, we employ a novel approach in collecting the summaries where, instead of having human annotators read a given paper, we ask them to read peer review comments about a paper and rewrite them into a TLDR. Peer reviews are written by domain experts and they often include a few sentences summarizing the paper, which facilitates our TLDR data collection and ensures high quality.

In addition to the unique technical challenges, TLDR generation has important real-world applications. Researchers are commonly faced with large collections of papers to process (e.g., a search result page or daily feed of new papers). Given the increasing pace of publication (Van Noorden, 2014), the ability to quickly discern a paper's key points and decide whether it's worth reading is critical for keeping up with the literature. Automatically-generated TLDRs could be a time-saving alternative to skimming abstracts.

We further present a method for generating TLDRs of scientific papers using BART (Lewis et al., 2019), a pretrained language model with strong performance on summarization. Drawing upon connections between TLDRs and a related task, title generation, we propose a multitask learning strategy using a title generation scaffold (Swayamdipta et al., 2018; Cohan et al., 2019) for improving TLDR generation for finetuning pretrained language models. We show that this method, while simple, is effective in extreme summarization for the scientific domain. Our contributions are summarized below:

1. We introduce the task of TLDR generation, a new form of extreme summarization in the scientific domain.
2. We release SciTLDR, a new dataset of 3,935 TLDRs including author-written summaries, and a test set that is additionally augmented with human-written summaries derived from the reviewer comments.
3. We propose a multitask learning strategy that exploits the related task of title generation to improve TLDR generation. We demonstrate its effectiveness against strong baselines using pretrained language models.

Category	Example phrase
Domain, field or area of study	*reinforcement learning, dependency parsing*
Problem of interest	*mode collapse, catastrophic forgetting*
Mode of contribution	*method, dataset, treebank, theorem*
General description of proposed method	*using graph convolution operations with dynamically computed graphs*
Main results or findings	*improved performance on ImageNet*
Value of work	*state-of-the-art, simple but effective*

Table 1: Example categories of information a TLDR might contain

2 Scientific paper TLDRs

A scientific paper TLDR is an extreme summary of a given scientific paper, conveying its main message or key contributions in one or two sentences (typically between 15 to 30 tokens in length). TLDRs are significantly less verbose than abstracts, allowing the reader to quickly understand what the paper is about and decide whether they want to continue reading it. TLDRs are already being used on some scholarly platforms. OpenReview[4] is one such example where authors are asked to submit TLDRs of their papers that communicate the main content of the paper to both reviewers and other interested scholars. TLDRs of scientific papers are also common on social media platforms, such as Twitter.

Table 1 provides a (non-exhaustive, non-disjoint) categorization of content that can appear in a TLDR, illustrated with example phrases taken from our dataset. We observe that there is often large variability in how TLDRs are written and any individual TLDR may contain only a subset of these aspects due to limited length. Further, what a paper's TLDR looks like or what information it should include is subjective and follows (community-specific) commonsense rather than a formally defined procedure.

For example, the TLDR *"A BERT model trained on scientific text"* omits the problem being solved and focuses on the method. But another TLDR *"A simple approach for preventing mode collapse in conditional GANs"* focuses on the problem being solved but omits the method. An alternative TLDR for the first example could explain the term BERT at a cost of being more verbose (e.g. *"A Transformer-based bi-directional language model*

[4]https://openreview.net/

Reviewer comment	The authors proposed a new clustering algorithm named deep continuous clustering (DCC) that integrates autoencoder into continuous clustering. As a variant of continuous clustering (RCC), DCC formed a global continuous objective for joint nonlinear dimensionality reduction and clustering. The objective can be directly optimized using SGD like method. Extensive experiments on image and document datasets show the effectiveness of DCC. However, part of experiments are not comprehensive enough. The idea of integrating autoencoder with continuous clustering is novel, and the optimization part is quite different. The trick used in the paper (sampling edges but not samples) looks interesting and seems to be effective. In the following, there are some detailed comments: 1. The paper is well written and easy to follow, except the definition of Geman-McClure function is...
Peer Review TLDR	Deep Continuous Clustering is a clustering method that integrates the autoencoder objective with the clustering objective then train using SGD.
Author-TLDR	A clustering algorithm that performs joint nonlinear dimensionality reduction and clustering by optimizing a global continuous objective.

Table 2: Example of a reviewer's comment rewritten as a TLDR

trained on scientific text"). But in this example, since the majority of readers who work in the same domain know about "BERT," explaining it again in the TLDR just adds more verbosity without providing any new information about the paper. Since TLDRs are inherently ultra-short, they are not necessarily self-contained statements, and understanding them requires some level of expertise within their respective scientific domain. Therefore, when designing our dataset, we assume readers have sufficient background knowledge to follow a general research topic in a given domain. This eliminates the need to include explanations for the common concepts or terms within a domain (e.g., "PennTreeBank," "LSTM," "BERT").

3 Dataset

We introduce SciTLDR, a multi-target dataset of 3,935 TLDRs of scientific articles in the Computer Science domain. The dataset is specific to the AI related papers that are hosted on OpenReview, a popular open publishing platform. SciTLDR includes a total of 3,229 papers each with at least one gold TLDR.

We use the OpenReview publishing platform as the basis of our data collections. Most papers on this platform are accompanied by an author-provided TLDR, making it a convenient data source. We use the OpenReview API [5] to collect pairs of papers and TLDRs, and access PDFs of papers. We then use the pipeline from S2ORC (Lo et al., 2020) to convert PDFs to structured, machine-readable full text. We use ScispaCy (Neumann et al., 2019) sentence segmentation for the baselines that require sentence boundaries.

Note that a small fraction of our dataset ($< 5\%$) did not have an available PDF file, so we could not parse their full body text. Although these papers only have abstracts, we still include them in the dataset, since one can generate a TLDR from an abstract alone. We split the dataset randomly into three sets of training, development, and test (with 60/20/20 ratios).

[5]https://github.com/openreview/openreview-py

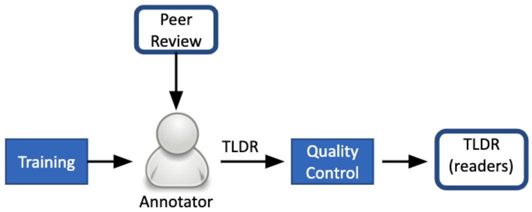

Figure 2: Annotation protocol for deriving TLDRs from peer reviews harvested from OpenReview.

3.1 Deriving TLDRs from peer reviews

There is often variability in human written summaries, as discussed in §2. Standard automated evaluation methods for summarization, such as the Rouge framework, rely on comparing system generated summaries with gold human-written summaries. Therefore, considering only one gold TLDR for each paper as a basis of evaluation, might result in inaccurate system quality assessment because content that might appear in a TLDR can have large variability (recall Table 1).

To address this limitation, we additionally collect a second type of gold summary for the papers in our test set. Compared with author written TLDRs, these TLDRs are from the reader's point of view. Having multiple targets allows us to capture

Dataset	# Docs	Doc length	Summ. length	Comp. ratio	Target
XSUM (2018)	226K	431	23	18.7	single
CNN (2015)	93K	760	46	16.1	single
DailyMail (2015)	220K	653	55	11.9	single
ArXiv (2018)	215K	4938	220	4.5	single
SciTLDR					
$\text{SciTLDR}_{\text{Full}}$	3.2k	5009	19	263.6	multi
$\text{SciTLDR}_{\text{Abst}}$	3.2k	159	19	8.3	multi
$\text{SciTLDR}_{\text{AIC}}$	3.2k	993	19	52.3	multi

Table 3: Comparison of datasets. The lengths are in tokens. Target summary can be single (only one gold summary per document) or multi (multiple gold summaries for each document). Compression ratio shows the ratio of document length to summary length. $\text{SciTLDR}_{\text{Full}}$ refers to SciTLDR with full-text while $\text{SciTLDR}_{\text{Abst}}$ only includes abstract and $\text{SciTLDR}_{\text{AIC}}$ includes abstract, introduction and conclusion sections.

the natural variation inherent in summarization and particularly in extreme summarization.

Writing TLDRs of scientific papers requires domain expertise and is challenging and expensive to crowdsource. To facilitate this and ensure a high-quality dataset, instead of writing a TLDR from the paper, we propose a new approach of writing a TLDR from peer reviews. This annotation approach makes it easier for annotators with sufficient background knowledge to write TLDRs without having to read the entire paper. Peer reviews often include one introductory paragraph with some sentences summarizing the key points in a paper. We ask annotators to extract the summary of the paper from a paper's review and rewrite it into a TLDR. Table 2 contains an example of a peer review and its corresponding TLDR next to the author-written TLDR. In addition, this approach provides a deeper level of abstraction, as the peer reviews already include summary statements of the paper and are rewritten into TLDRs.

For this task, we hire 12 annotators, at a rate of $20 USD per hour.[6] The annotators have experience reading natural language processing papers and we provide them with an hour of training to learn how to complete the task. We then manually check the annotations for factual correctness and revise or discard the lower quality TLDRs accordingly. This process is depicted in Figure 2. We obtain peer review comments from OpenReview, which publishes reviews for certain conferences (e.g., ICLR). We then use these comments as inputs for our human annotation process.

[6]This is in line with the median hourly wage in the Washington State at the time of writing.

	1-gram	2-gram	3-gram
XSUM	35.76	83.45	95.50
CNN	16.75	54.33	72.42
DailyMail	17.03	53.78	72.14
Arxiv	17.5	48.1	71.4
SciTLDR			
$\text{SciTLDR}_{\text{Full}}$	5.65	21.57	62.60
$\text{SciTLDR}_{\text{Abst}}$	25.99	65.75	79.62
$\text{SciTLDR}_{\text{AIC}}$	11.30	47.60	68.54

Table 4: Percentage of n-grams in summary that are not in source document. For SciTLDR we show analysis of different sections as the source document.

3.2 Dataset Analysis

Table 3 provides statistics of our SciTLDR dataset compared with existing summarization datasets. There are a few key differences to note. First, SciTLDR is a smaller dataset than existing datasets. Extreme summaries of scientific papers are significantly more difficult and expensive to collect because they require domain expertise to write and assess for quality. In contrast, other datasets such as XSUM, were collected automatically from the web.

The next notable difference is the length of the documents and summaries. Table 3 compares the average summary and document length of SciTLDR with other existing datasets. The compression ratio (ratio of document length to TLDRs) is significantly higher compared to existing datasets, making it a more challenging summarization dataset. It is possible for models to only consider some sections of the paper as input (e.g., abstract or abstract + introduction + conclusion). This reduces the total document length and makes the task more manageable for models, at the cost of potentially missing some important information from other sections of the paper. We therefore define $\text{SciTLDR}_{\text{Abst}}$ as a variant of the dataset that only has abstract as input while $\text{SciTLDR}_{\text{AIC}}$ has abstract, introduction and conclusion sections, and in $\text{SciTLDR}_{\text{Full}}$, the entire paper is considered as input. Even when limiting the input to the abstract, introduction, and conclusion sections, the compression ratio of SciTLDR is still significantly higher than other existing summarization datasets.

Another critical difference is that SciTLDR is multi-target. SciTLDR has no fewer than two gold TLDRs for each paper in the test set, one written by the authors (Author-TLDR) and the others manually written from peer review comments (PR-TLDR) as described in §3.1. The standard evaluation measure in summarization is Rouge (Lin, 2004), which quantifies lexical overlaps between a system generated and a gold summary as a score for evaluating quality. In scientific TLDRs, we observe high lexical variability in human-written summaries for the same paper, while all these summaries are perfectly valid.

This is evidenced by the low Rouge overlap between Author-TLDRs and PR-TLDRs. Comparing Author-TLDRs and PR-TLDRs achieves a Rouge-1 score of only 27.4. Furthermore, PR-TLDRs overlap less with the titles than Author-TLDRs. Comparing the TLDRs to the paper titles, Author-TLDRs receive a Rouge-1 of 34.1 while PR-TLDRs only receive a Rouge-1 of 24.7. This stresses the importance of the multi-target evaluation setting in extreme summarization, as a single paper can have multiple, sufficiently different TLDRs.

4 TLDRGEN

We propose TLDRGEN, a novel method for learning to generate TLDRs. Our approach addresses two main challenges: (1) the limited size of the training data and (2) the need for domain knowledge in order to produce high quality TLDRs. To address these challenges, we propose utilizing *titles* of scientific papers as additional generation targets. As titles often contain key information about a paper, we hypothesize that training a model to generate titles will allow it to learn how to locate salient information in the paper that will be also useful for generating TLDRs. In addition, all papers have a title, and thus we have an abundant supply of paper-title pairs for training. Following previous success utilizing control codes (Keskar et al., 2019), we append control codes specifying the style of the output to the source documents and perform training in a multitask manner by shuffling paper-title pairs with paper-TLDR pairs.

We utilize an additional dataset of approximately 20K paper-title pairs from arXiv in the computer science domain for the title generation task. [7] We then up-sample SCITLDR to match the size of the title dataset. We shuffle SCITLDR and the arXiv data together, appending each source with control tags ⟨|TLDR|⟩ and ⟨|TITLE|⟩, respectively. This allows the parameters of the model to learn to generate TLDRs and titles at the same time, taking advantage of the close relationship between the two tasks. To test if both the multitask setup and the control codes are important to this setup, we also conduct the following experiments: (1) the same multitask setup but without the control codes and (2) pretraining on title generation before finetuning on SCITLDR (§6).

5 Experiments

In this section we discuss our experimental setup, baselines that we use for SCITLDR, as well as implementation and training details.

5.1 Input space

Previous studies have found that the most salient information in a paper for writing a summary is often found in the abstract, introduction, and conclusion sections (Sharma et al., 2019). In our analysis of SCITLDR, we also found that an oracle extractive model that uses the full-text of the paper only slightly improves over a model that uses abstract, introduction, and conclusion sections (see §6). This further suggests that most salient information relevant to TLDR generation exists in these sections. Therefore, we hypothesize that a model should be able to learn to generate TLDRs by relying mainly on these sections. This simplifies our experimental setup both for our approach and the baselines.[8]

The abstract, alone, is another reasonable input space. From a practical application point of view, production level systems do not always have access to the full-text of the paper. In these cases, having the ability to generate a TLDR solely from the abstract would be useful. Also, considering this additional experimental setting allows us to compare quality of TLDRs using these two separate input spaces.

5.2 Baselines

We evaluate the task of TLDR generation with both extractive and abstractive state-of-the-art summarization baselines.

Extractive methods We consider both unsupervised and supervised extractive methods. For our unsupervised extractive baseline, we focus on the

[7]This includes all papers on arXiv with at least one of the following tags CS.CL, CS.CV, CS.LG, CS.AI, CS.NE, and STAT.ML *and* have identified introduction and conclusion sections in S2ORC (Lo et al., 2020). S2ORC is a dataset that provides structured full text for academic papers, including those in arXiv.

[8]Most pretrained Transformer models have an input length constraint and will not scale well for long documents.

Method	SciTLDR$_{\text{Abst}}$			SciTLDR$_{\text{AIC}}$		
	Rouge-1	Rouge-2	Rouge-L	Rouge-1	Rouge-2	Rouge-L
Ext. Oracle	47.7	24.7	38.5	52.4	29.0	42.9
PACSUM (Zheng and Lapata, 2019)	22.5	7.5	17.4	23.0	7.4	17.8
BERTSUMEXT (Liu and Lapata, 2019)	36.7	15.8	29.4	28.3	10.4	23.5
BART (Lewis et al., 2019)	41.6	19.8	33.8	42.0	20.5	34.9
BART.XSUM (Lewis et al., 2019)	41.2	19.7	33.4	43.2	**21.2**	35.6
TLDRGen (Ours)	**42.6**	**21.0**	**34.8**	**43.4**	21.0	**35.6**

Table 5: Main results of TLDR generation comparing our method with extractive and abstractive baselines. In SciTLDR$_{\text{Abst}}$ the input is the abstract, while in SciTLDR$_{\text{AIC}}$ the input to the models is abstract+introduction+conclusion sections. All baselines are finetuned on SciTLDR, except for PACSUM, our unsupervised baseline.

	Rouge-1	Rouge-2	Rouge-L
SciTLDR$_{\text{Abst}}$	47.7	24.7	38.5
SciTLDR$_{\text{AIC}}$	52.4	29.0	42.9
SciTLDR$_{FullText}$	54.5	30.6	45.0

Table 6: Sentence-level extractive oracle results

state-of-the-art unsupervised summarization model, PACSUM (Zheng and Lapata, 2019). PACSUM is an extension of TextRank (Mihalcea and Tarau, 2004), utilizing directed edges and a pretrained BERT model as a sentence encoder. Additionally, we include a state-of-the-art supervised extractive summarization baseline, BERTSumExt (Liu and Lapata, 2019). This model uses BERT as a sentence encoder augmented with inter-sentence Transformer layers to capture interactions between sentences.

Abstractive methods Since TLDRs often include information that is spread across multiple sentences, we hypothesize that abstractive summarization models would be strong baselines for this task. We particularly focus on BART-large, a state-of-the-art pretrained model with strong results in text generation and summarization (Lewis et al., 2019). Additionally, we utilize BART finetuned on XSUM, hypothesizing that the task of extreme summarization on news articles might transfer to scientific papers when further finetuning on SciTLDR. For brevity, we will refer to these models as BART and BART.XSUM, respectively. We use the Fairseq (Ott et al., 2019) implementation of the BART model.

5.3 Training and implementation details

All our experiments are done on Titan V or V100 GPUs. Below, we describe training hyperparameters for the baselines, as well as our multitask finetuning approach. Hyperparameters for the baselines and our method are selected based on best performance on the validation set.

Baselines BERTSumExt is trained with a batch size of 3000 sentences per batch and for 50000 total steps. We use a learning rate of 2e-3 and a dropout rate of 0.1. For SciTLDR$_{\text{Abst}}$, the document is truncated at 512 tokens, and for SciTLDR$_{\text{AIC}}$, 1500 tokens. For the BART finetuning experiments, we train all the models for 500 steps and 100 warmup steps, for an approximate training time of 45 minutes. We use a learning rate 3e-5, an update frequency of 1, and a max tokens per batch of 1024.

TLDRGen Our implementation is based on the Fairseq framework (Ott et al., 2019). On SciTLDR$_{\text{Abst}}$, we train our TLDRGen model for 11,000 total steps and on SciTLDR$_{\text{AIC}}$, we train for 45,000 total steps. For both, we used a learning rate 3e-5, a linear warmup and linear decay learning rate scheduler with 20% warmup steps proportion.

5.4 Evaluation

We use the current standard evaluation metric in summarization, the Rouge (Lin, 2004) framework. Following recent work on summarization (Narayan et al., 2018; Lewis et al., 2019), we use Rouge-1, Rouge-2, and Rouge-L as our automatic metrics. The main difference with existing datasets is that we use multi-target summaries as references. That is, for each paper, we calculate Rouge score of the system-generated TLDR with respect to each of the gold TLDRs individually, and then take the maximum Rouge score over these gold TLDRs as

Method	SciTLDR$_{\text{Abst}}$			SciTLDR$_{\text{AIC}}$		
	Rouge-1	Rouge-2	Rouge-L	Rouge-1	Rouge-2	Rouge-L
BART.XSUM → SciTLDR	41.2	19.7	33.4	43.2	**21.2**	35.6
→ arXiv titles → SciTLDR	17.7	2.7	13.9	21.9	2.2	17.8
→ arXiv titles + SciTLDR (no codes)	40.5	19.5	33.5	42.6	21.0	34.8
→ Title + SciTLDR (with codes)	**42.6**	**21.0**	**34.8**	**43.4**	21.0	**35.6**

Table 7: Alternative training configurations. Arrows "→" indicate finetuning. Plus signs "+" indicate multitask training with shuffled data. The final model is TLDRGen.

the final Rouge score for that paper. An alternative approach to aggregate scores would be to take the average. However, we argue that taking the maximum, rather than average, over the targets for each prediction is appropriate for the task of TLDR generation. That is because, given the variability in human written TLDRs, a good TLDR generation model should be able to generate a TLDR that is close to one of the human written TLDRs, and not necessarily close to all of them. For example, the gold TLDRs shown in Table 2 are lexically different while both are perfectly valid. Therefore, if a system generates a TLDR that is lexically very similar to one of the TLDRs, it should not be penalized for not being similar to the other gold TLDR.

5.5 Oracle

We include a sentence-level extractive oracle as an upper-bound of Rouge performance for the methods. This oracle captures the maximum extractive Rouge score possible given a gold TLDR. We select a single sentence in the document with the highest Rouge-1 overlap for each gold TLDR and output that as the oracle TLDR.[9]

6 Results

Table 5 shows the main results comparing all baselines with our proposed method over the two input spaces, abstract (SciTLDR$_{\text{Abst}}$) and abstract+introduction+conclusion (SciTLDR$_{\text{AIC}}$).

Oracle The oracle score gives an upper-bound on the performance on sentence-level extraction. As shown in Table 6, by increasing the input space from SciTLDR$_{\text{Abst}}$ to SciTLDR$_{\text{AIC}}$, the oracle achieves, on average, approximately 5 points of Rouge-1 improvement, indicating that the introduction and conclusion sections include additional information helpful for generating TLDRs. However, there is a smaller performance boost when increasing the input space from SciTLDR$_{\text{AIC}}$ to SciTLDR$_{FullText}$. This supports our hypothesis that the most salient information to a TLDR is in the abstract, introduction, and conclusion.

Baselines As shown in Table 5, we observe that BART finetuned on SciTLDR achieves a high performance, even when finetuned on a small amount of data. Furthermore, finetuning BART on XSUM appears to give performance boosts to SciTLDR$_{\text{AIC}}$ but not SciTLDR$_{\text{Abst}}$. We also observe that the extractive methods achieve Rouge scores closer to the oracle on SciTLDR$_{\text{Abst}}$. Despite the high oracle score on SciTLDR$_{\text{AIC}}$, the extractive methods generally do not perform well. In the case of PACSUM, we find that PACSUM heavily prefers to extract sentences from the abstract, even when provided with additional context. Furthermore, we can see that BERTSUMEXT performs worse on SciTLDR$_{\text{AIC}}$ than SciTLDR$_{\text{Abst}}$. This is likely because there is not enough data for the model to properly learn which sentences to extract.

TLDRGen Our TLDRGen method (described in §4) outperforms all other baselines. Specifically, we observe 1 to 1.4 points in Rouge improvement over the best next baseline models. This indicates the value of the title generation task and our proposed multitask framework in generating TLDRs. Furthermore, to test the utility of both the multitask setting and the control codes, we conduct similar title-based experiments but (1) without the control codes and (2) finetuning on titles rather than utilizing the multitask setting, before further finetuning on SciTLDR. As shown in Table 7, title only finetuning performs poorly. We can also see that, while the multitask model without control codes performs well, it does not outperform BART.XSUM fine-

[9]Including an abstractive oracle is not meaningful as an abstractive oracle trivially always gets Rouge score of 1.0 by generating the gold TLDR. While it is possible for abstractive methods to outperform the extractive oracle, we consider the extractive oracle as a reasonable performance upper-bound for both the extractive and abstractive the methods.

Gold (Author)	A comparison of five deep neural network architectures for detection of malicious domain names shows surprisingly little difference.
Gold (Peer Review)	Authors propose using five deep architectures for the cybersecurity task of domain generation algorithm detection.
BART.XSUM $\rightarrow$SCITLDR$_{\mathrm{Abst}}$	In this paper, we present an empirical comparison of the effectiveness of different deep learning architectures for the detection of Domain Name Corruption (DGA).
BART.XSUM $\rightarrow$SCITLDR$_{\mathrm{AIC}}$	In this paper, we compare the performance of five different deep learning architectures for character based text classification.
TLDRGEN$_{\mathrm{Abst}}$	We compare various deep neural networks for character based text classification.
TLDRGEN$_{\mathrm{AIC}}$	We compare five different deep neural network architectures for character based text classification of domain names generated by malware.

Table 8: Example of generated and gold TLDRs. Arrows "$\rightarrow$" indicate finetuning.

tuned on SCITLDR. This is likely because the model is unable to differentiate between the two tasks without the control codes. Finally, TLDRGEN, multitask training with control codes, outperforms all of our baselines and alternative configurations of the title training. This serves as evidence that both the multitask learning strategy and the use of control codes is important to the performance of TLDRGEN.

Table 8 shows a randomly sampled example for which each BART model variant generates a different prediction. We observe that most of the generated TLDRs are sensible with subtle differences which highlights the domain knowledge required to analyze the TLDRs. Looking at the difference between the TLDRGEN$_{\mathrm{Abst}}$ and TLDRGEN$_{\mathrm{AIC}}$ we observe that TLDRGEN$_{\mathrm{AIC}}$ includes more phrases that are in the gold TLDRs, a trend that is we also commonly see throughout the entire dataset. This suggests that the TLDRGEN$_{\mathrm{AIC}}$ model with access to abstract, introduction and conclusion sections is better able to find salient information from various input locations compared with the abstract-based TLDRGEN$_{\mathrm{Abst}}$.

7 Related work

Extreme summarization This work builds on previous work in both scientific document summarization and extreme summarization. All previous work in extreme summarization has been in the news domain. To our knowledge, the only other existing dataset is XSUM, introduced by (Narayan et al., 2018). XSUM consists of news articles and single sentence summaries. Our work introduces an extreme summarization dataset in the scientific domain. Additionally, various pretraining methods of transformer models have been found to be successful in extreme summarization (Zhang et al., 2019; Lewis et al., 2019; Liu and Lapata, 2019; Bi et al., 2020). While our multitask and domain transfer approaches are built upon BART (Lewis et al., 2019), a state-of-the-art generation model, our setup is general and can be applied to any other pretrained models.

Scientific document summarization In the scientific domain, previous work has focused on longer summaries. Existing datasets include the arXiv dataset introduced by (Cohan et al., 2018), an abstract generation task, and SciSummNet, introduced by (Yasunaga et al., 2019), which includes human annotated reference summaries. Previous work has also leveraged additional context to aid in summary generation, such as citation context (Qazvinian and Radev, 2008; Cohan and Goharian, 2017) or recordings of conference talks (Lev et al., 2019). Other work has also attempted to generate citation contexts, given two papers (Luu et al., 2020). By introducing the task of TLDR generation, our work bridges the gap in existing summarization efforts in the scientific domain that mostly focus on generating long summaries.

8 Conclusion

We propose the task of TLDR generation in the scientific domain. We introduce SCITLDR, a new dataset of scientific TLDRs including author-written summaries, and a test set that is additionally augmented with summaries derived from reviewer comments. We propose a multitask learning approach which improves TLDR generation using a title generation scaffold task when finetuning pretrained language models, and show that our approach outperforms strong baselines.

Although our results are promising, future work can leverage the full text of the paper in the methodology, capturing more context of the paper. Furthermore, we make multiple assumptions about the background knowledge of the reader. A potential future direction is to work on explicitly modeling the readers background knowledge, creating personalized TLDRs. Additionally, our annotation process of converting peer review comments to TLDRs can be applied to a number of sources that contain summaries of a paper. For example, future work can leverage the wide use of scientific TLDRs on social media, such as Twitter and Reddit, to build a more informal version of the dataset. Finally, our model is currently trained on English-language papers in the computer science domain. Future work can expand to other languages and scientific domains.

References

Bin Bi, Chenliang Li, Chen Wu, Ming Yan, and Wei Wang. 2020. Palm: Pre-training an autoencoding and autoregressive language model for context-conditioned generation. *ArXiv*, abs/2004.07159.

Arman Cohan, Waleed Ammar, Madeleine van Zuylen, and Field Cady. 2019. Structural scaffolds for citation intent classification in scientific publications. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 3586–3596, Minneapolis, Minnesota. Association for Computational Linguistics.

Arman Cohan, Franck Dernoncourt, Doo Soon Kim, Trung Bui, Seokhwan Kim, Walter Chang, and Nazli Goharian. 2018. A discourse-aware attention model for abstractive summarization of long documents. In *NAACL-HLT*.

Arman Cohan and Nazli Goharian. 2017. Scientific article summarization using citation-context and article's discourse structure. *arXiv preprint arXiv:1704.06619*.

Donna Harman and Paul Over. 2004. The effects of human variation in DUC summarization evaluation. In *Text Summarization Branches Out*, pages 10–17, Barcelona, Spain. Association for Computational Linguistics.

Karl Moritz Hermann, Tomas Kocisky, 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.

Nitish Shirish Keskar, Bryan McCann, Lav R. Varshney, Caiming Xiong, and Richard Socher. 2019. Ctrl: A conditional transformer language model for controllable generation. *ArXiv*, abs/1909.05858.

Guy Lev, Michal Shmueli-Scheuer, Jonathan Herzig, Achiya Jerbi, and David Konopnicki. 2019. Talksumm: A dataset and scalable annotation method for scientific paper summarization based on conference talks. In *ACL*.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Ves Stoyanov, and Luke Zettlemoyer. 2019. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *ArXiv*, abs/1910.13461.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Yang Liu and Mirella Lapata. 2019. Text summarization with pretrained encoders. In *EMNLP/IJCNLP*.

Kyle Lo, Lucy Lu Wang, Mark Neumann, Rodney Kinney, and Daniel S. Weld. 2020. S2orc: The semantic scholar open research corpus. In *Proceedings of ACL*.

Kelvin Luu, Rik Koncel-Kedziorski, Kyle Lo, Isabel Alyssa Cachola, and Noah A. Smith. 2020. Citation text generation. *ArXiv*, abs/2002.00317.

Rada Mihalcea and Paul Tarau. 2004. Textrank: Bringing order into texts. In *EMNLP 2004*.

Shashi Narayan, Shay B. Cohen, and Mirella Lapata. 2018. Dont give me the details, just the summary! topic-aware convolutional neural networks for extreme summarization. pages 1797–1807.

Mark Neumann, Daniel King, Iz Beltagy, and Waleed Ammar. 2019. Scispacy: Fast and robust models for biomedical natural language processing.

Myle Ott, Sergey Edunov, Alexei Baevski, Angela Fan, Sam Gross, Nathan Ng, David Grangier, and Michael Auli. 2019. fairseq: a fast, extensible toolkit for sequence modeling. In *Proceedings of NAACL-HLT 2019: Demonstrations*.

Vahed Qazvinian and Dragomir R. Radev. 2008. Scientific paper summarization using citation summary networks. In *Proceedings of the 22Nd International Conference on Computational Linguistics - Volume 1*, COLING '08, pages 689–696, Stroudsburg, PA, USA. Association for Computational Linguistics.

Eva Sharma, Chen Li, and Lu Wang. 2019. Bigpatent: A large-scale dataset for abstractive and coherent summarization. In *ACL*.

Swabha Swayamdipta, Sam Thomson, Kenton Lee, Luke Zettlemoyer, Chris Dyer, and Noah A. Smith. 2018. Syntactic scaffolds for semantic structures. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 3772–3782, Brussels, Belgium. Association for Computational Linguistics.

Richard Van Noorden. 2014. Global scientific output doubles every nine years. *Nature news blog*.

Oleg Vasilyev, Tom Grek, and John Bohannon. 2019. Headline generation: Learning from decomposable document titles.

Michihiro Yasunaga, Jungo Kasai, Rui Zhang, Alexander Richard Fabbri, Irene Li, Dan Friedman, and Dragomir R. Radev. 2019. Scisummnet: A large annotated corpus and content-impact models for scientific paper summarization with citation networks. In *AAAI*.

Klaus Zechner. 1996. Fast generation of abstracts from general domain text corpora by extracting relevant sentences. In *Proceedings of the 16th conference on Computational linguistics-Volume 2*, pages 986–989. Association for Computational Linguistics.

Jingqing Zhang, Yao Zhao, Mohammad Saleh, and Peter J. Liu. 2019. Pegasus: Pre-training with extracted gap-sentences for abstractive summarization. *ArXiv*, abs/1912.08777.

Hao Zheng and Mirella Lapata. 2019. Sentence centrality revisited for unsupervised summarization. In *ACL*.