# Toward Structured Related Work Generation with Novelty Statements

**Kazuya Nishimura**[1], **Kuniaki Saito**[2], **Tosho Hirasawa**[2], **Yoshitaka Ushiku**[2]
[1] National Cancer Center Japan
[2]OMRON SINIC X Corp.

## Abstract

To help readers understand the novelty and the research context, an excellent related work section is structured (*i.e.,* the section consists of multiple paragraphs organized by several research topics) and includes descriptions of novelty. However, previous studies viewed related work generation as just multi-document summarization, and the structure and novelty statements are ignored. In this paper, we redefine the related work generation task as structured related work generation with novelty statement (**STRoGeNS**), and we propose datasets and automatic evaluation metrics for structure and novelty for STRoGeNS. On our structured related work generation with novelty statement, we investigate the state-of-the-art language models and deliver insights about the effects of pre-training and input context. In addition, we confirm the validity of the proposed automatic evaluation metrics with human evaluation. [1]

## 1 Introduction

A related work section of a scientific article plays an important role in helping readers understand the context of the research field and the novelty of their work (Hoang and Kan, 2010). The well-written related work is properly structured (*i.e.,* make paragraphs with topics determined from the relations between current work and previous works) and clearly described novelty. To write such a related work section, authors must categorize a number of articles into several topics, summarize them, and highlight the novelty compared to them, which requires a deep and broad understanding of the research field, and is, of course, a very time-consuming process.

Given the difficulty, the automatic related work generation has been proposed (Hoang and Kan, 2010; Hu and Wan, 2014). However, existing studies viewed the related work generation as multi-document summarization and do not sufficiently consider the structure of the related work (i.e., whether paragraphs are organized based on research topics reflected by the target research field) and whether it states the novelty of the current work. For example, current abstractive related work studies aim to generate a single paragraph (Lu et al., 2020; Chen et al., 2021, 2022) or citation sentence of related work (Mao et al., 2022), and the outputs of them assume a single research topic and do not require structuring. While Liu et al. (2022) aimed to generate structured summarization, the target is a literature review constructed not with research topics but with components of the document such as background and method, and the objective is not always to highlight novelty.

**Input**

# Target article information $t$
**Title:** SCOTT: Self-Consistent Chain-of-Thought Distillation
**Abstract:** Existing neural models have difficulty generalizing.…

# a set of cited articles $\mathcal{C}$
[1] Title:{Title}, Abstract: {Abstract}
[2] Title: {Title}, Abstract: {Abstract}
:
[$n$] Title: {Title}, Abstract: {Abstract}

**Output:** Related work section (whole section)

**Free-text Rationales** A variety of datasets have been proposed to collect human-annotated rationales alongside each task instance **[1]**, aiming to train the downstream models to …

**Prompted Self-Rationalization Models** Recent works have been proposed to prompt large LMs to generate a free-text rationale before ….**[6]** and fail to faithfully represent the underlying reasoning process **[4]**. In contrast, our student is trained to be more faithful towards its generated rationales using a smaller LM.

**Knowledge Distillation** There exist some works that explore the idea of distilling rationales knowledge from….

Figure 1: Overview of STRoGeNS. Given target article $t$ and a set of cited articles $\mathcal{C}$, our task aims to output a related work section with multiple paragraphs determined based on research topics with citations. The number of the cited article is used as an identifier (*i.e.,* [#i]) in the target text. Red text indicates a novelty statement.

[1]We will release our dataset and scripts upon publication.



*Proceedings of the Fourth Workshop on Scholarly Document Processing (SDP 2024)*, pages 38–57

In this paper, we redefine related work generation not just as summarization but as a STructured Related Work Generation with Novelty Statement (**STRoGeNS**) and propose dataset and evaluation metrics. Here, we define a structured related work as a section in which each paragraph is organized by research topics, and the articles related to the topic of the paragraph are cited and summarized in the paragraph.

Our contributions to generate STRoGeNS are as follows.

- We propose a novel related work generation task, namely structured related work generation with novelty statement (**STRoGeNS**), that aims to generate a related work section with multiple paragraphs organized by research topics and descriptions of novelty statements.
- We propose large-scale datasets and automatic evaluation metrics for structured related work generation with novelty statement. The datasets focus on the trade-off of quantity and quality, and the evaluation metrics are designed to evaluate structure and novelty.
- We evaluate state-of-the-art summarization models on our STRoGeNS task and find two challenges: the text generation beyond summarization and the capability of utilizing long contexts.

## 2 Related work

**Extractive related work generation.** The prototype of automated related work section generation has been proposed by Hoang and Kan (2010). They viewed the task as a topic-biased summarization and proposed an extraction method by selecting sentences based on given topics. Inheriting this setting, several extractive summarization methods have been proposed by Hu and Wan (2014); Chen and Zhuge (2019); Wang et al. (2020b, 2018); Deng et al. (2021).

**Abstractive related work generation.** The first trial of abstractive related work generation is citation sentence generation (AbuRa'ed et al., 2020; Xing et al., 2020), and afterward, a paragraph of related work has also attracted attention (Lu et al., 2020; Chen et al., 2021, 2022). For example, Chen et al. (2022) has proposed target-aware related work generation to generate target-centric related work. Liu et al. (2023a) has improved quality and readability with the causality invention module. There are discussions about the task of the related work generation. Shi et al. (2023) has proposed a task that combines reference retrieval and related work generation. Funkquist et al. (2023) unified previous datasets and defined related work generation as cited generation task.

However, these studies viewed related work generation as a summarization task and did not focus on the structure and novelty of related work. Unlike previous studies, we aim to generate structured related work with novelty statements.

**Automatic evaluation.** Several $n$-gram-based evaluations have been proposed to measure the overlap of the gold standard and generated text, such as BLEU (Papineni et al., 2002), ROUGE (Lin, 2004). To reflect the context of a sentence, embedding-based evaluations, which measure the similarity of the embedding vectors extracted from the text by the NLG model, have been proposed, such as (Yuan et al., 2022; Zhao et al., 2019). Recently, evaluations using a large language model (LLM) have been proposed (Liu et al., 2023b; Fu et al., 2023). Unlike $n$-gram and embedding-based evaluation, this evaluation reflects the evaluation criteria by prompts and allows for evaluation across various criteria, such as coherence, consistency, and fluency. This paper verified unexplored novelty evaluation with the LLM-based evaluation.

**Task-specific evaluation.** Since the generation aims differ depending on the target task, evaluation metrics are studied for each downstream task. For example, Durmus et al. (2020); Wang et al. (2020a) focused on consistency of summary, Ye et al. (2021); Ghazarian et al. (2022) focused on the coherence of the dialogue response generation, and Funkquist et al. (2023); Gao et al. (2023); Rashkin et al. (2021) focused on citations of generated text. However, the evaluation of related works has not been considered. To the best of my knowledge, this is the first trial to evaluate the novelty and structure of related work.

## 3 Structured Related work generation with Novelty statements (STRoGeNS)

We propose a new challenge, namely STructured Related Work Generation with Novelty Statement (**STRoGeNS**), for related work generation. STRoGeNS is motivated by two mandatory properties

| Dataset | Pairs | Words (input) | Words (Output) | Input Doc (Num) | #Para. |
|---|---|---|---|---|---|
| Multi-XScience (Lu et al., 2020) | 40,528 | 778.1 | 116.4 | 4.4 | 1 |
| S2ORC (Chen et al., 2021) | 136,655 | 1067.4 | 148.7 | 5.0 | 1 |
| Delve (Chen et al., 2021) | 78,927 | 622.6 | 228.6 | 3.7 | 1 |
| TAS2 (Chen et al., 2022) | 117,700 | 1036.0 | 134.8 | 4.8 | 1 |
| TAD (Chen et al., 2022) | 218,255 | 1071.4 | 162.3 | 5.2 | 1 |
| BigSurvey-MDS (Liu et al., 2022) | 4,478 | 11,893.1 | 1,051.7 | 76.3 | 1 |
| SciReviewGen (Kasanishi et al., 2023) | 10,130 | 11,734.4 | 7,193.8 | 68.1 | 1 |
| STRoGeNS-arXiv22 | 85,853 | 3,046.2 | 514.3 | 16.6 | 4.22 |
| STRoGeNS-conf22 | 15,079 | 3,669.1 | 508.5 | 20.4 | 4.27 |
| STRoGeNS-conf23 | 4,762 | 4,836.6 | 504.6 | 25.7 | 4.04 |

Table 1: Statistics and feature of relevant datasets.

(structure and novelty statement) of related work for improving readers' understanding and writers' inspiration.

### 3.1 Task definition

Our STRoGeNS is defined as follows: Given target article information $\mathbf{t}$ and a set of $n$ cited articles $\mathcal{C} = \{\mathbf{c}_i\}_i^n$, the model generates related work section $R$ of the target article, which $R$ consists of $m$ paragraphs $\mathbf{p}_1, ..., \mathbf{p}_m$ [2]. When the $\mathbf{c}_i$ is cited in $\mathbf{p}_i$, [#i] is used as identifier. Figure 1 shows the example of the input and output of our task.

Unlike previous abstractive related work generation task (Liu et al., 2022; Funkquist et al., 2023), which outputs a single paragraph of related work as a summary, our task outputs a related work section with structure and novelty statements in addition to a summary.

### 3.2 Dataset

**Construction.** Collecting samples with high-quality structure and accurate novelty statements is necessary to generate our target. On the other hand, a sufficient number of articles is required to understand a wide range of research fields. We address this trade-off between quality and quantity by creating a quantity-oriented dataset (**STRoGeNS-arXiv22**) and a quality-oriented dataset (**STRoGeNS-conf22**). Additionally, we create a dataset for test (**STRoGeNS-conf23**).

**STRoGeNS-arXiv22:** To generate a quantity-oriented dataset, we collect articles that include related works from 1,882,082 articles of unarXiv 2022 (Saier et al., 2023), a large-scale dataset of arXiv[3]. With fully structured text and citations of unarXiv 2022[4], we extract titles, abstracts, sections with associated or previous work tags, and titles of cited articles. Since the abstracts of cited articles are not contained in unarXiv 2022, we retrieve the abstract using Semantic Scholar API (Ammar et al., 2018). As a result, 130,585 related works with citation information are collected. Then, samples with only one paragraph ($m = 1$) or more than eight paragraphs ($m > 8$) or less than five cited articles ($n < 5$) were excluded. Finally, we collected 85,853 articles. Here, all citation identifiers are replaced with [#i]. Although this dataset is ensured in quantity, it may contain many low-quality structured sections and inaccurate novelty statements because it contains articles that have not been peer-reviewed.

To generate a quality-oriented dataset, we generate two datasets for training and test from top-tier peer-reviewed proceedings. First, we collected PDFs of conference proceedings from conference proceedings websites. Then, the PDFs are converted to structured markdown-styled text with Neural Optical Understanding for Academic Documents (NOUGAT) (Blecher et al., 2023), which is a transformer-based optical character recognition model. The title of the cited article is obtained by matching the identifiers in references with the identifiers in the paragraph based on the citation identifier such as [#i] or Author, et al., where regular expression operations are used to extract the citation identifiers. Similar to STRoGeNS-arXiv22,

[2]While our dataset includes paper IDs that allow access to the full text, we only use titles and abstracts for inputs ($\mathbf{t}$ and $\mathbf{c}_i$) in terms of input length and ease of data collection.

[3]We could not use S2ROC dataset (Lo et al., 2020), a large-scale annotated scientific dataset, as it was poorly structured and difficult to retrieve paragraphs accurately.

[4]unarXiv 2022 parsed the LaTeXsource file and has JSON-styled text with paragraphs, sections, and citation links.

| Category | #article |
|---|---|
| Computer Science | 73.8 % |
| Physics | 1.67 % |
| Mathematics | 6.19 % |
| Statistics | 13.7 % |
| Quantitative Biology | 0.41 % |
| Quantitative Finance | 0.14 % |
| Electrical Engineering and Systems Science | 3.93 % |
| Economics | 0.12 % |

Table 2: Distribution of category in STRoGeNS-arXiv22.

| Conferences | #article |
|---|---|
| ACL | 7.61 % |
| EMNLP | 12.1 % |
| NAACL | 4.61 % |
| CVPR | 35.9 % |
| ECCV | 9.00 % |
| ICCV | 13.7 % |
| ICLR | 9.62 % |
| ICML | 7.58 % |

Table 3: Distribution of conferences in STRoGeNS-conf22.

| Conferences | #article |
|---|---|
| ACL | 11.9% |
| CVPR | 37.5% |
| ICCV | 37.2% |
| ICML | 13.4% |

Table 4: Distribution of conferences in STRoGeNS-conf23.

| Dataset | $r_{\mathrm{SE}}$ | $r_{\mathrm{PE}}$ | $r_{SR}$ |
|---|---|---|---|
| STRoGeNS-arXiv22 | 18.8% | - | 81.2% |
| STRoGeNS-conf22 | 12.8% | 9.27% | 77.9% |
| STRoGeNS-conf23 | 11.4% | 6.90% | 81.7% |

Table 5: Quality analysis of datasets (%). SE, PE, and SR indicate the rate of retrieval errors of Semantic Scholar, parse errors by the Nougat, and successfully obtained citations.

cleaning is performed based on the number of paragraphs and the number of citations articles, and the abstracts are parsed by Semantic scholar API.

**STRoGeNS-conf22.** This dataset focuses on the quality of articles and is used for training. We collected 28,211 PDFs from CVPR, ECCV, ICCV, ACL, EMNLP, NAACL, ICML, and ICLR published before 2022. Finally, we collected 15,079 structured related work sections from the PDFs using the above steps.

**STRoGeNS-conf23.** We use the article published after 2023 for the test. We collected 7,248 PDFs from four conferences: ACL, CVPR, ICCV, and ICML, and 5,578 related works were collected by above steps. Since some articles may have been uploaded to arXiv before being accepted at the conferences, we remove samples by manually checking the title to avoid data leakage. First, fuzzy matching is performed between the titles of STRoGeNS-arXiv22 and STRoGeNS-conf23, and articles with a match score over 90 are collected. Then, we manually checked the 534 collected articles and removed 409 articles. Finally, 4,762 structured related work sections are corrected by parsing PDFs.

**Statistics.** Table 1 shows statistics of our datasets and related previous datasets. Unlike previous datasets for single-paragraph generation, our dataset averages about four paragraphs. Although our datasets aim to generate an entire related work section, they are comparable in size to S2ORC and TAS2.

Table 2 shows the distribution of the category of samples in STRoGeNS-arXiv22. Since related work is often described in computer science articles, 73.8 % of articles are in computer science. Table 3, and 4 show distribution of conference in STRoGeNS-conf22 and STRoGeNS-conf23. Due to the explosive increase in the number of papers on computer vision, the proportion of CVPR, ICCV, and ECCV documents is large. Although we use the proposed datasets without considering the category in this trial, the proposed dataset has tags of category and conference, so it can also be used for studies that take the category into account.

**Quality.** Our dataset contains two retrieval errors of citation information: a retrieval error caused by Semantic Scholar API (SE) and a parse error of NOUGAT (PE). Table 5 shows the rate of SE, PE, and the successful retrieval SR. Our datasets successfully retrieved about 80% of the abstract of citations on average.

### 3.3 Automatic Evaluation

We evaluate the generated text of models from three viewpoints: **Summarization** whether the generated contains meaningful information, **Structure** whether the citations within the paragraphs are accurately grouped, and **Novelty statement** whether the generated section contains a novelty statement.

**Summarization.** We use a ROUGE (Lin, 2004), widely used as an automatic evaluation metric in summarization. The metrics measure the $n$-gram overlap between a generated text and a gold stan-

dard. We use ROUGE-1.5.5 implementation of (Lin, 2004) with "-n 2" option, and input is whole sections of the gold standard and generated text.

**Structure.** Paragraphs of well-structured related works should cite articles according to the topics determined from the target research field. In our test data, most related work sections are expected to follow the above requirements because they are collected from peer-reviewed top-tier proceedings. Therefore, we design structure metrics based on whether the citation trends in the paragraph are similar to the citation trends in the gold standard section.

We propose four metrics: (1) F1, (2) ARI, (3) ARI' and (4) $\mathrm{MAE_p}$.

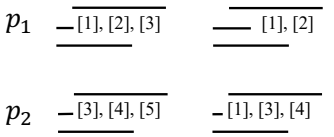
The F1 measures the citation similarity among most similar paragraphs of the gold standard and generated text. F1-based metrics are defined as follows:

$$\mathrm{F1} = \frac{1}{m}\sum_{i=1}^{m}\max_{j}\mathrm{F1}(f(\mathbf{p}_i), f(\hat{\mathbf{p}}_j)), \quad (1)$$

where $f$ is a function that extracts a citation identifier, $p_i$ is $i$-th paragraph in the gold standard, $\hat{p}_j$ is $j$-th paragraph in the generated section, and F1 is F1-score calculation. Since the metric is calculated between the most similar paragraphs, the paragraphs with low similarity are ignored, and the entire paragraphs cannot be measured.

We propose a metric that introduced the Adjusted Rand Index (ARI) (Steinley, 2004; Hubert and Arabie, 1985), a clustering metric to evaluate citation trends of low-similarity paragraphs. As shown in Figure 2, citation trends could be evaluated with a clustering evaluation manner by treating the paragraphs of the original and generated section as clusters. By applying clustering, changes in the number of paragraphs can be treated in the same way as changes in the number of clusters, and thus, citation trends for the whole section can be evaluated. We calculate the Adjusted Rand Index (ARI) with assigned cluster IDs of the gold standard and generated section.

As shown in Figure 2, citations may be used in multiple paragraphs, unlike clustering. Then, some articles will not have corresponding IDs, as shown in the unmatched citation (UC) in Figure 2. To address this problem, we modify the Adjusted Rand Index to account for unmatched pairs (UC) caused by multi-paragraphs. RI is formulated as $\mathrm{RI} = \frac{\#\mathrm{TPair}}{\#\mathrm{Pair}}$, where #Pair and #TPair are the total




| Ref. ID | Cluster ID | |
|---|---|---|
| | Gol. | Gen. |
| [1] | 1 | 1 |
| [2] | 1 | 1 |
| [3] | 1 | UC |
| [1] | UC | 2 |
| [3] | 2 | 2 |
| [4] | 2 | 2 |
| [5] | 2 | MC |

Figure 2: Example of structure evaluation with ARI. The gold standard and generated related work sections have two paragraphs with citations. Then, cluster ID is assigned for each reference, as shown in the right table. UC and MC indicate an unmatched citation and a missed citation, respectively.

number of pairs and the number of pairs that have correct relation. We add the number of unmatched samples (#UC) to the denominator of RI. Then, modified $\mathrm{RI}'$ are formulate as follows:

$$\mathrm{RI}' = \frac{\#\mathrm{TPair}}{\#\mathrm{Pair} + \#\mathrm{UC}}. \quad (2)$$

Adjusted Rand Index is calculated with modified $RI'$ as follows: $\mathrm{ARI}' = \frac{\mathrm{RI}' - \mathrm{ExpectedRI}}{\max(\mathrm{RI}) - \mathrm{ExpectedRI}}$. This metric is calculated like regular ARI when there are no unmatched citations, and the score decreases with unmatched citations.

In addition, we calculate the mean absolute error with the number of paragraphs in reference and generated ($\mathrm{MAE}_p$).

Finally, we use four metrics F1, ARI, ARI', $\mathrm{MAE}_p$ for structure evaluation.

**Novelty statement.** To evaluate whether the generated text contains a novelty statement, we used the prompt-based evaluation method using LLM (Liu et al., 2023b). It has been reported that when used for evaluation, LLM correlates better with human evaluation than traditional automated evaluation metrics in terms of consistency, conciseness, and grammar. However, whether it is a valid index for novelty evaluation has not been explored.

We follow G-Eval (Liu et al., 2023b), which combines chain-of-thoughts (CoT) and a form-filling paradigm for LLM-based evaluation. G-Eval consists of two steps: evaluation step generation and score prediction. First, we manually write an initial prompt $\mathbf{a}_{\mathrm{init}}$ with a task instruction and evaluation criteria. Then, GPT generates evaluation steps. Next, we create a prompt $a$ with task instruction and evaluation criteria, evaluation steps, and

related work section to judge, and output format (the prompt is shown in appendix B.). Finally, the Novelty score $s_n \in \{0, 1\}$ is obtained from GPT prediction $s_n = \mathbf{GPT}(\mathbf{a})$.

The model is trained to use all cited articles in this setting. However, sometimes generated text misses cite all articles, as shown in [5] in Figure 2. Since the above metrics ignore such missing citations, we also evaluate the rate of missing citation $r_{MC}$.

## 4 Experiments

### 4.1 Experimental setting

**Model.** We evaluated the performance of various extractive and abstractive summarization models on our dataset:

**TextRank (Mihalcea and Tarau, 2004) and LexRank (Erkan and Radev, 2004):** graph-based unsupervised extractive summarization methods using the page rank technique and centrality scoring, respectively.

**GPT3.5 and GPT4 (Brown et al., 2020):** A largea language model released by Openai. We used the latest GPT-3.5 Turbo model (`gpt-3.5-turbo-1106`) and GPT4 (`gpt-4-0613`) for generation.

**BART (Lewis et al., 2019):** A famous encoder and decoder transformer pre-trained with autoregressive and denoising tasks.

**PEGASUS (Zhang et al., 2020):** A encoder and decoder transformer pre-trained for summarization with Gap Sentences Generation and Masked Language Model.

**LED (Beltagy et al., 2020):** A Longformer-Encoder-Decoder (LED) is an encoder and decoder transformer that introduces local and global attention instead of self-attention to utilize long context.

**Llama2-7B (Touvron et al., 2023):** An open-source large language model developed by Meta. The model has a 4096 context length with seven billion parameters and is trained in an autoregressive manner.

In this comparison, we fine-tuned abstractive models first pre-trained model with STRoGeNS-arxiv22 for 20 epochs and then fine-tuned with STRoGeNS-conf22 for 20 epochs.

**Implementation.** The models were trained using the AdamW optimizer with a learning rate of $3e-5$. The learning rate was adjusted using a linear scheduler with warm-up. For other parameters, we used the default parameter of huggingface implementation. All the models are trained on NVIDIA A100 80GB with batch size $= 64$. To match the max token length of the abstractive models, we truncated each abstract of the cited articles.

The implementation details for each model are as follows. We added a "/n" token to the PEGASUS tokenizer to generate paragraphs. LED is fine-tuned with 8,192 token length. For Llama2-7B, we used LoRA (Hu et al., 2021) with $r = 64$, $\alpha = 32$, dropout $= 0.1$ parameters.

### 4.2 Results

**Evaluation with summarization and structure metrics.** Table 6 compares the model in terms of summarization and structure metrics comparisons. The (+digit or -digit) indicates the increase or decrease in the performance from only fine-tuned with STRoGeNS-conf22 to this condition. Table 7 shows the performance of each model fine-tuned on STRoGeNS-conf22 (C) or STRoGeNS-arxiv22 (A) in terms of R-1, F1, and $r_{MC}$ (full results are shown in Appendix D). Overall, BART achieved the best performance in terms of both metrics. We summarize the points of the results below.

**Evaluation with novelty metrics.** Table 8 shows the average novelty score for each method. Due to budget constraints, we verified the novelty score with GPT4 (`gpt-4-0613`) on 100 randomly sampled test samples. The generated texts of each model were input to the GPT with our prompts, and novelty scores were estimated. It shows that the text generated by BART contains a larger number of novelty statements than that of other models. 0.91 is a good trend since about 10% of original articles do not contain novelty statements. LED and Llama2 can handle long contexts, but the proportion of novelty statements decreases. This suggests that our task requires a novelty-centered model rather than simply using long inputs.

**Pre-training for summarization does not contribute to improving performance.** The performance of PEGASUS is significantly inferior to BART on both metrics. The main difference between PEGASUS and BART lies in their pre-training approaches; PEGASUS specializes in summarization tasks, whereas BART is designed for general generation tasks. Therefore, the characteristics of our task differ from simple summarization tasks, suggesting that it involves difficulty in summarization, citations, and structure ability.

| Method | Summarization | | | Structure | | | | Other |
|---|---|---|---|---|---|---|---|---|
| | R-1 ↑ | R-2 ↑ | R-L ↑ | F1 ↑ | ARI ↑ | ARI' ↑ | $MAE_p$ ↓ | $r_{MC}$ ↓ |
| TextRank | 31.6 | 7.5 | 24.7 | - | - | - | - | - |
| LexRank | 31.5 | 8.19 | 21.6 | - | - | - | - | - |
| GPT3.5 | 38.2 | 7.7 | 28.5 | 69.0 | 39.6 | 33.9 | 2.12 | 56.7% |
| GPT4* | 41.4 | 7.8 | 29.8 | 66.7 | 32.1 | 29.8 | 1.73 | 41.2 % |
| BART | **47.4** (+4.9) | **13.1** (+1.3) | **32.7** (+1.7) | 69.1 (+6.6) | **55.1** (+15.5) | **54.0** (+15.6) | 2.03 (+0.07) | 5.42% (-0.38) |
| PEGASUS | 25.4 (+7.3) | 6.2 (+1.2) | 19.1 (+5.6 | 48.8 (+17.6) | 11.3 (-0.8) | 11.0 (+5.4) | 4.04 (+0.68) | 21.2% (+9.0) |
| LED | 44.0 (+2.5) | 11.8 (+0.5) | 29.2 (+0.7) | 59.5 (+3.4) | 35.2(+9.1) | 33.7 (+9.1) | 2.18(-0.1) | **5.0 %** (-0.4) |
| Llama2-7B | 42.7 (+0.1) | 9.0(+0.1) | 29.4(+0.2) | **69.8**(-0.2) | 48.7(+0.4) | 47.3 (+0.4) | **1.66** ±0 | 15.9% (-0.9) |

Table 6: Comparisons on each model in terms of summarization and structure. The methods above the dashed line are unsupervised methods, and below the supervised methods. (+digit or -digit) indicates the increase or decrease in the performance from only fine-tuned with STRoGeNS-conf22 to this condition. Text colors red and blue indicate positive and negative effects, respectively. *: Due to budget constraints, we generated related work with GPT4 using only 1500 samples; thus, the results are for reference purposes.

**Models accepting long context fail to utilize the rich context of cited articles.** Surprisingly, the performances of LED and Llama2 are inferior to BART. Since we truncated the abstract of cited articles for max token length, BART can only use the first few sentences, and LED can use all sentences (The example of input text is shown in Appendix C). Although LED and Llama2 can use a rich context of the research field, models failed to utilize the context to improve performance. Since LED achieved great performance on other long text summarization (Liu et al., 2022, 2023a), our task requires understanding the relationships between articles from a long text, and it seems that the ability to process long scientific papers, which are different from traditional tasks, is insufficient.

## 5 Discussion

### 5.1 Model analysis

**Quality vs. Quantity.** As shown in Table 7, pre-training under STRoGeNS-arxiv22 is more effective than STRoGeNS-conf22 on models with short input token lengths (BART and PEGASUS). On the other hand, the performance under condition STRoGeNS-arxiv22 is inferior to condition STRoGeNS-conf22 on long input token length models (LED and Llama2). When using long context, the variation of inputs increases, and it becomes more difficult to process long input by understanding the meaning than short input lengths. Then, the long context model could not leverage the benefit of arxiv22 which has large variations and low-quality articles. The result suggests that to train models with longer input lengths, the dataset's quality becomes more crucial than for those with shorter inputs.

| Method | D | R-1 ↑ | F1 ↑ | $r_{MC}$ ↓ |
|---|---|---|---|---|
| BART | C | 42.5 | 62.5 | 5.8% |
| | A | **46.1** | 66.4 | 2.8% |
| PEGASUS | C | 18.1 | 31.2 | 12.0% |
| | A | 17.7 | 40.2 | 6.0% |
| LED | C | 41.5 | 56.1 | **5.4 %** |
| | A | 40.9 | 12.3 | 95.0% |
| Llama2-7B | C | 42.6 | **70.0** | 16.8% |
| | A | 37.6 | 9.0 | 95.2% |

Table 7: Comparisons on each dataset. C: fine-tuning with STRoGeNS-conf22, A: fine-tuning with STRoGeNS-arxiv22.

**Validity of novelty statements.** As shown in Table 8, the generated section of BART contained the highest number of novelty statements compared to other models. However, since the input of BART is truncated (see Appendix C), it may not deeply understand the research field. In this study, we assessed whether the generated section contains novelty statements, but evaluating the validity of these statements is necessary for future work.

**The potential of large language models.** While GPT3.5 contains many missed citations, it achieves comparable performance with fine-trained Llama2 on structure metrics. Llama2 achieved the best performance in terms of structure when the fine-tuned models on STRoGeNS-conf22 (Table 7). These results suggest that under conditions of a small amount of training data, LLM exhibits superior structuring capabilities compared to other models.

| Method | Novelty score |
|---|---|
| GPT3.5 | 0.17 |
| BART | **0.91** |
| LED | 0.53 |
| Llama2 | 0.56 |

Table 8: Average novelty score of each model for 100 samples. The score is estimated by GPT in a G-EVAL manner.

| Method | $r$ | $\rho$ | $\tau$ |
|---|---|---|---|
| F1 | 47.2 | 56.4 | 31.9 |
| ARI | 33.4 | 31.5 | 20.8 |
| ARI' | 38.3 | 35.7 | 24.9 |
| $\mathrm{MAE}_p$ | -41.6 | -45.8 | -32.3 |

Table 9: Correlation between human evaluation with structure metrics. r, $\rho$, $\tau$ indicate Pearson's, Spearman's, and Kendall's correlation, respectively.

On the other hand, despite Llama2 having a wider range of academic knowledge than other models, the total performance is inferior to BART (Table 6). The reason is that the Llama2 does not fully leverage the benefits of quantity-oriented STRoGeNS-arxiv22. Therefore, we should explore methods that can utilize the advantages of both large models and datasets.

The concern with the language model is the number of missing citations. GPT, which is attracting attention for its generation of citations (Gao et al., 2023), fails citations for over 50 % of the articles on zero-shot settings on zero-shot condition. Even using GPT4, the rate of missed citations is still high. The rate of missed citation of fine-tuned Llama2 is 15%, and it is larger than BART.

### 5.2 Meta-evaluation

We conducted human evaluations to run meta-evaluation of structure and novelty metrics.

**Structure metrics.** We evaluated the structure of the generated section by Llama2 for 25 randomly selected samples from STRoGeNS-conf2023. Four annotators have manually inspected the structure score by comparing the generated section with the gold standard based on the following five criteria: (1) All topics of the paragraph do not match, (2) most topics of the paragraph do not match, (3) the topics of almost half do not match, (4) the few topics in the section do not match, (5) all topics in the section match.

Table 9 shows a correlation between human evaluation with our proposed metrics (F1, ARI, ARI', and $\mathrm{MAE}_p$) in terms of Pearson's (r), Spearman's ($\rho$), and Kendall's ($\tau$). The Fleiss' kappa (Warrens, 2010) among annotators is 0.156 and shows slight agreements. F1, ARI, and ARI' show a positive correlation, and $\mathrm{MAE}_p$ show a negative correlation with human evaluations. ARI', the extended version of ARI, shows a higher correlation than ARI, and it demonstrates the effectiveness of the proposed extension. The lower $\mathrm{MAE}_p$, which is a difference in the number of paragraphs, is better, but it does not reflect the meaning within the paragraphs. In this human evaluation, a single score is given to one related work; hence, reflecting on each paragraph's quality is difficult. While ARI' focuses on each paragraph, F1 focuses only on the highest similarity paragraphs. Therefore, F1 seems to show a higher correlation. While not reflected in human evaluation, we argue that the quality of each paragraph is also important, and it is desirable to use both ARI' and F1.

**Novelty metrics.** We randomly selected 100 samples from the STRoGeNS-conf23 dataset and annotated them with sentence-level spans of novelty statements, finding 90 samples have novelty statements. Additionally, we synthesize the samples lack of novelty statements by removing annotated novelty statements from 90 samples. Then, we asked GPT4 (gpt-4-0613) used in our novelty metrics to determine whether each of the 190 samples contains novelty statements or not. As a result of evaluating 190 samples of novelty statements using GPT, the accuracy was 92%. This result supports that GPT is capable to judge the presence of novelty statements in related work, and consequently the validity of the novelty metrics used in our task.

## 6 Conclusion

In this paper, we redefined automatic related work generation as STructured Related Work Generation with Novelty Statement (**STRoGeNS**). For STRoGeNS, we proposed datasets focusing on the trade-off of quality and quantity and automatic evaluation metrics for structure and novelty. Using the quantity-oriented and quality-oriented datasets, we evaluated the performance of state-of-the-art models and showed that the task is not solved by simple summarization ability, and the ability to handle input length is required. In addition, we conducted human evaluation and demonstrated the effectiveness of the proposed automated evaluation metrics.

## Limitations

Our dataset for STRoGeNS is mainly covered English and the computer science category. It contains Mathematics and Mathematics, but the test data only covers conferences of computer science. We should extend to the cross-domain and cross-lingual for the future.

We evaluated structure on the assumption that the citations of generated text are somewhat correct. However, as discussed in the discussion of citation evaluation (Funkquist et al., 2023), some citations do not take their meaning into account. Ideally, we should have factual consistency of citation. To carry this point, we should build a model that can evaluate the factual consistency of scientific articles using citation text datasets such as (Mao et al., 2022; Li et al., 2022).

In this paper, related work sections were just generated using an off-the-shelf language model. As in the early research (Hoang and Kan, 2010; Hu and Wan, 2014), when considering the entire section, a method that focuses on the topics handled within the section is required. We should consider using topic relationships between paragraphs rather than just generating entire sections with LLM.

## Ethics Statement

Our datasets consist of articles from the scientific field, and we do not foresee a negative societal impact. As with text generation, the model tuned with our dataset may have risk to output factually inconsistent and biased texts.

## Acknowledgements

This work is supported by JST Moonshot R&D Program Grant Number JPMJMS2236. We used the computational resources of AI Bridging Cloud Infrastructure (ABCI) provided by National Institute of Advanced Industrial Science and Technology (AIST).

## References

Ahmed Ghassan Tawfiq AbuRa'ed, Horacio Saggion, Alexander V. Shvets, and Àlex Bravo. 2020. Automatic related work section generation: Experiments in scientific document abstracting. *Scientometrics*, 125:3159 – 3185.

Waleed Ammar, Dirk Groeneveld, Chandra Bhagavatula, Iz Beltagy, Miles Crawford, Doug Downey, Jason Dunkelberger, Ahmed Elgohary, Sergey Feldman, Vu Ha, Rodney Kinney, Sebastian Kohlmeier, Kyle Lo, Tyler Murray, Hsu-Han Ooi, Matthew Peters, Joanna Power, Sam Skjonsberg, Lucy Lu Wang, Chris Wilhelm, Zheng Yuan, Madeleine van Zuylen, and Oren Etzioni. 2018. Construction of the literature graph in semantic scholar. In *North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*, pages 84–91.

Iz Beltagy, Matthew E. Peters, and Arman Cohan. 2020. Longformer: The long-document transformer. *ArXiv*, abs/2004.05150.

Lukas Blecher, Guillem Cucurull, Thomas Scialom, and Robert Stojnic. 2023. Nougat: Neural optical understanding for academic documents. *ArXiv*, abs/2308.13418.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems (NeurIPS)*, pages 1877–1901.

Jingqiang Chen and Hai Zhuge. 2019. Automatic generation of related work through summarizing citations. *Concurrency and Computation: Practice and Experience*.

Xiuying Chen, Hind Alamro, Li Mingzhe, Shen Gao, Rui Yan, Xin Gao, and Xiangliang Zhang. 2022. Target-aware abstractive related work generation with contrastive learning. *Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval*.

Xiuying Chen, Hind Alamro, Li Mingzhe, Shen Gao, Xiangliang Zhang, Dongyan Zhao, and Rui Yan. 2021. Capturing relations between scientific papers: An abstractive model for related work section generation. In *Association for Computational Linguistics (ACL)*.

Zekun Deng, Zixin Zeng, Weiye Gu, Jiawen Ji, and Bolin Hua. 2021. Automatic related work section generation by sentence extraction and reordering. In *Workshop on AI + Informetrics (AII)*.

Esin Durmus, He He, and Mona Diab. 2020. FEQA: A question answering evaluation framework for faithfulness assessment in abstractive summarization. In *Association for Computational Linguistics (ACL)*.

Günes Erkan and Dragomir R Radev. 2004. LexRank: Graph-based lexical centrality as salience in text summarization. *Journal of artificial intelligence research*, 22:457–479.

Jinlan Fu, See-Kiong Ng, Zhengbao Jiang, and Pengfei Liu. 2023. GPTScore: Evaluate as you desire. *ArXiv*, abs/2302.04166.

Martin Funkquist, Ilia Kuznetsov, Yufang Hou, and Iryna Gurevych. 2023. CiteBench: A benchmark for scientific citation text generation. In *Empirical Methods in Natural Language Processing (EMNLP)*.

Tianyu Gao, Howard Yen, Jiatong Yu, and Danqi Chen. 2023. Enabling large language models to generate text with citations. In *Empirical Methods in Natural Language Processing (EMNLP)*.

Sarik Ghazarian, Nuan Wen, Aram Galstyan, and Nanyun Peng. 2022. DEAM: Dialogue coherence evaluation using AMR-based semantic manipulations. In *Association for Computational Linguistics*.

Cong Duy Vu Hoang and Min-Yen Kan. 2010. Towards automated related work summarization. In *International Conference on Computational Linguistics (Coling)*.

J. Edward Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, and Weizhu Chen. 2021. LoRA: Low-rank adaptation of large language models. *ArXiv*, abs/2106.09685.

Yue Hu and Xiaojun Wan. 2014. Automatic generation of related work sections in scientific papers: An optimization approach. In *Empirical Methods in Natural Language Processing (EMNLP)*.

Lawrence J. Hubert and Phipps Arabie. 1985. Comparing partitions. *Journal of Classification*, 2:193–218.

Tetsu Kasanishi, Masaru Isonuma, Junichiro Mori, and Ichiro Sakata. 2023. SciReviewGen: A large-scale dataset for automatic literature review generation. In *Findings of Association for Computational Linguistics: ACL 2023*.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdel rahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2019. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Association for Computational Linguistics (ACL)*.

Xiangci Li, Biswadip Mandal, and Jessica Ouyang. 2022. CORWA: A citation-oriented related work annotation dataset. In *North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT)*.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Association for Computational Linguistics (ACL)*.

Jiachang Liu, Qi Zhang, Chongyang Shi, Usman Naseem, Shoujin Wang, Liang Hu, and Ivor Tsang. 2023a. Causal intervention for abstractive related work generation. In *Findings of the Association for Computational Linguistics: EMNLP 2023*.

Shuaiqi Liu, Jiannong Cao, Ruosong Yang, and Zhiyuan Wen. 2022. Generating a structured summary of numerous academic papers: Dataset and method. In *International Joint Conference on Artificial Intelligence (IJCAI)*.

Yang Liu, Dan Iter, Yichong Xu, Shuohang Wang, Ruochen Xu, and Chenguang Zhu. 2023b. G-eval: NLG evaluation using gpt-4 with better human alignment. In *Empirical Methods in Natural Language Processing (EMNLP)*.

Kyle Lo, Lucy Lu Wang, Mark Neumann, Rodney Kinney, and Daniel Weld. 2020. S2ORC: The semantic scholar open research corpus. In *Association for Computational Linguistics (ACL)*.

Yao Lu, Yue Dong, and Laurent Charlin. 2020. Multi-XScience: A large-scale dataset for extreme multi-document summarization of scientific articles. In *Empirical Methods in Natural Language Processing (EMNLP)*.

Yuning Mao, Ming Zhong, and Jiawei Han. 2022. CiteSum: Citation text-guided scientific extreme summarization and domain adaptation with limited supervision. In *Empirical Methods in Natural Language Processing (EMNLP)*.

Rada Mihalcea and Paul Tarau. 2004. TextRank: Bringing order into text. In *Empirical Methods in Natural Language Processing (EMNLP)*.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Association for Computational Linguistics (ACL)*.

Hannah Rashkin, Vitaly Nikolaev, Matthew Lamm, Michael Collins, Dipanjan Das, Slav Petrov, Gaurav Singh Tomar, Iulia Turc, and D. Reitter. 2021. Measuring attribution in natural language generation models. *Computational Linguistics*, 49:777–840.

Tarek Saier, Johan Krause, and Michael Färber. 2023. unarXive 2022: All arXiv Publications Pre-Processed for NLP, Including Structured Full-Text and Citation Network. In *Joint Conference on Digital Libraries (JCDL)*.

Zhengliang Shi, Shen Gao, Zhen Zhang, Xiuying Chen, Zhumin Chen, Pengjie Ren, and Zhaochun Ren. 2023. Towards a unified framework for reference retrieval and related work generation. In *Findings of the Association for Computational Linguistics: EMNLP 2023*.

Douglas L. Steinley. 2004. Properties of the hubert-arabie adjusted rand index. *Psychological methods*, 9 3:386–96.

Hugo Touvron, Louis Martin, Kevin R. Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Daniel M. Bikel, Lukas Blecher, Cristian Cantón Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony S. Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel M. Kloumann, A. V. Korenev, Punit Singh Koura, Marie-Anne Lachaux,

Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, R. Subramanian, Xia Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zhengxu Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. 2023. Llama 2: Open foundation and fine-tuned chat models. *ArXiv*, abs/2307.09288.

Alex Wang, Kyunghyun Cho, and Mike Lewis. 2020a. Asking and answering questions to evaluate the factual consistency of summaries. In *Association for Computational Linguistics (ACL)*.

Pancheng Wang, Shasha Li, Haifang Zhou, Jintao Tang, and Ting Wang. 2020b. ToC-RWG: Explore the combination of topic model and citation information for automatic related work generation. *IEEE Access*, 8:13043–13055.

Yongzhen Wang, Xiaozhong Liu, and Zheng Gao. 2018. Neural related work summarization with a joint context-driven attention mechanism. In *Empirical Methods in Natural Language Processing (EMNLP)*.

Matthijs J Warrens. 2010. Inequalities between multi-rater kappas. *Advances in data analysis and classification*, 4:271–286.

Xinyu Xing, Xiaosheng Fan, and Xiaojun Wan. 2020. Automatic generation of citation texts in scholarly papers: A pilot study. In *Association for Computational Linguistics (ACL)*.

Zheng Ye, Liucun Lu, Lishan Huang, Liang Lin, and Xiaodan Liang. 2021. Towards quantifiable dialogue coherence evaluation. In *Association for Computational Linguistics and International Joint Conference on Natural Language Processing (ACL-IJCNLP)*.

Weizhe Yuan, Graham Neubig, and Pengfei Liu. 2022. BARTScore: Evaluating generated text as text generation. In *Advances in Neural Information Processing Systems (NeurIPS)*.

Jingqing Zhang, Yao Zhao, Mohammad Saleh, and Peter J. Liu. 2020. PEGASUS: Pre-training with extracted gap-sentences for abstractive summarization. In *International Conference on Machine Learning (ICML)*.

Wei Zhao, Maxime Peyrard, Fei Liu, Yang Gao, Christian M. Meyer, and Steffen Eger. 2019. MoverScore: Text generation evaluating with contextualized embeddings and earth mover distance. In *Empirical Methods in Natural Language Processing and International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*.

You will be given one related work and title and abstract written for a computer science paper. Your task is to rate the related work on several metrics.

# target paper info
Title, abst.
…
# references
[1] {title}, {abst}
…
[N] {title}, {abst}

# Output format
Output format:
Output is related work section that consists of 3~8 paragraphs with novelty statement. Ensure that there is a line break between each paragraph. When citing references, use the format '[reference number]' for citations. Enclose related work contents with <>.

Figure 3: Prompt for related work generation with GPT.

## A Novel $n$-gram of dataset.

To measure how abstracted the dataset is, we report the proportion of novel $n$-grams in the target-related work section. Table 12 shows the $n$-grams for each dataset. Our dataset trends are similar to Bigsurvey and SciReviewGen, which aims to generate a literature review.

## B Prompts for GPT

Figure 3 shows the prompt for GPT to generate related work.

Figure 4 shows our prompt for novelty evaluation. First, Evaluation steps **s** are generated by GPT. Then, we obtain the novelty score by inputting the whole prompt.

Table 11 shows the performance of Llama2-7B at each conference. Although our dataset contains a lot of computer vision articles, we can not confirm any major differences in performance.

## C Example of input for each model

The default maximum token lengths of BART, Llama2, and LED are 1,024, 4,096, and 16,384. The average token length of our dataset is about 6,000. Therefore, we set the length for LED to 8,192 We set the maximum output token length of Llama2 as 640, so the input length of Llama2 is 3456.

# Task introduction **i**
You will be given one related work, title, and abstract written for a computer science paper. Your task is to rate the related work on the following metric. Please make sure you read and understand these instructions carefully. Please keep this document open while reviewing, and refer to it as needed.

# Evaluation criteria **e**
Evaluation Criteria:
Novelty statement (0 or 1) - whether related work contains novelty statement.

# Evaluation steps **s** (GPT generated)
**1. Read the Related Work Section Thoroughly:** Begin by reading the entire related work section carefully. This will give you a complete understanding of the context and how the authors have positioned their work in relation to existing research.
**2. Identify the Novelty Statement:** Look specifically for a statement or a set of statements where the authors articulate what is new or different about their work compared to the existing literature. This can often be found towards the end of the related work section, but it might also be interspersed throughout the section.
**3. Evaluate the Novelty Statement:**
Presence (0 or 1): Determine if there is a clear statement of novelty. If such a statement exists, score it as '1'. If there is no explicit or implicit statement that outlines what makes the paper's contribution new or unique, score it as '0'.

# Sample to judge **d**
Related work: {Related work}

# Output format **f**
Output format:

Output only rate with JSON format.

Novelty statement: {}

Figure 4: Prompt for novelty evaluation.

Figure 5, 6, and 7 shows example of input length of each model. BART input only contains a few sentences of abstract for each cited article, Llama2 contains about 80% of abstract, and LED contains almost all abstract, on average.

## D Result of fine-tuning on each dataset

Table 10 shows the full performance of each model on fine-tuned with WREWD-arxiv22 for 20 epochs (A) and WREWD-conf22 for 20 epochs (C). For BART and PEGASUS with a maximum token length of 1,024, the summarization and structure performance is better than the conditions using the STRoGeNS-conf22 dataset. On the other hand, LED and Llama2, which can use longer contexts, perform worse than conditions using the STRoGeNS-conf22 dataset.

## E Examples of generated results of BART and Llama2

Figure 8 and 9 shows examples of input and output. Both BART and LLAMA results have paragraphs organized by research topics and contain a novelty statement.

BART (1024 token)

Target paper:
Title: Curvature-Aware Training for Coordinate Networks
Abstract: Coordinate networks are widely used in computer vision due to their ability to represent signals as compressed, continuous entities. However, training these networks with first-order optimizers can be slow, hindering their use in real-time applications. Recent works have opted for shallow voxel-based representations to achieve faster training, but this sacrifices memory efficiency. This work proposes a solution that leverages second-order optimization methods to significantly reduce training times for coordinate networks while maintaining their compressibility. Experiments demonstrate the effectiveness of this approach on various signal modalities, such as audio, images, videos, shape and neural radiance fields (NeRF).

[1] CoIL: Coordinate-based Internal Learning for Imaging Inverse Problems, Abstract: We propose Coordinate-based Internal Learning (CoIL) as a new deep-
[2] Implicit Neural Representations with Periodic Activation Functions, Abstract: Implicitly defined, continuous, differentiable signal representations parameterized by neural networks have emerged as a
[3] Learning Implicit Fields for Generative Shape Modeling, Abstract: We advocate the use of implicit fields for learning generative models of shapes and introduce an implicit field decoder
[4] Local Deep Implicit Functions for 3D Shape, Abstract: The goal of this project is to learn a 3D shape representation that enables accurate surface reconstruction, compact storage,
[5] The Implicit Bias of Minima Stability: A View from Function Space, Abstract: The loss terrains of over-parameterized neural networks have multiple global min
[6] DeepSDF: Learning Continuous Signed Distance Functions for Shape Representation, Abstract: Computer graphics, 3D computer vision and robotics communities have produced multiple approaches to representing 3D
[7] Nerfies: Deformable Neural Radiance Fields, Abstract: We present the first method capable of photorealistically reconstructing deformable scenes using photos/videos captured
[8] D-NeRF: Neural Radiance Fields for Dynamic Scenes, Abstract: Neural rendering techniques combining machine learning with geometric reasoning have arisen as one of the most promising approaches for
[9] DeRF: Decomposed Radiance Fields, Abstract: With the advent of Neural Radiance Fields (NeRF), neural networks can now render novel views of a 3
[10] PIFu: Pixel-Aligned Implicit Function for High-Resolution Clothed Human Digitization, Abstract: We introduce Pixel-aligned Implicit Function
[11] NeRF-: Neural Radiance Fields Without Known Camera Parameters, Abstract: Considering the problem of novel view synthesis (NVS) from only a set of 2D images
[12] pixelNeRF: Neural Radiance Fields from One or Few Images, Abstract: We propose pixelNeRF, a learning framework that predicts a continuous neural scene representation conditioned on
[13] NICE-SLAM: Neural Implicit Scalable Encoding for SLAM, Abstract: Neural implicit representations have recently shown encouraging results in various domains, including promising progress
[14] Factor Fields: A Unified Framework for Neural Fields and Beyond, Abstract: We present Factor Fields, a novel framework for modeling and representing signals. Factor Fields decomposes a signal
[15] Optimizing Neural Networks with Kronecker-factored Approximate Curvature, Abstract: We propose an efficient method for approximating natural gradient descent in neural networks
[16] Adaptive, Limited-Memory BFGS Algorithms for Unconstrained Optimization, Abstract: The limited-memory BFGS method (L-BFGS)
[17] Stochastic L-BFGS: Improved Convergence Rates and Practical Acceleration Strategies, Abstract: We revisit the stochastic limited-memory Broyden–F
[18] A Linearly-Convergent Stochastic L-BFGS Algorithm, Abstract: We propose a new stochastic L-BFGS algorithm and prove a
[19] Shampoo: Preconditioned Stochastic Tensor Optimization, Abstract: Preconditioned gradient methods are among the most general and powerful tools in optimization. However,
[20] Efficient Full-Matrix Adaptive Regularization, Abstract: Adaptive regularization methods pre-multiply a descent direction by a preconditioning matrix. Due

Figure 5: Example of truncated input text for BART.

LLAMA2 (4096 – 640=3456)

Input: Target paper:
Title: Curvature-Aware Training for Coordinate Networks
Abstract: Coordinate networks are widely used in computer vision due to their ability to represent signals as compressed, continuous entities. However, training these networks with first-order optimizers can be slow, hindering their use in real-time applications. Recent works have opted for shallow voxel-based representations to achieve faster training, but this sacrifices memory efficiency. This work proposes a solution that leverages second-order optimization methods to significantly reduce training times for coordinate networks while maintaining their compressibility. Experiments demonstrate the effectiveness of this approach on various signal modalities, such as audio, images, videos, shape and neural radiance fields (NeRF).

[1] CoIL: Coordinate-based Internal Learning for Imaging Inverse Problems, Abstract: We propose Coordinate-based Internal Learning (CoIL) as a new deep-learning (DL) methodology for the continuous representation of measurements. Unlike traditional DL methods that learn a mapping from the measurements to the desired image, CoIL trains a multilayer perceptron (MLP) to encode the complete measurement field by mapping the coordinates of the measurements to their responses. CoIL is a self-supervised method that requires no training examples besides the measurements of the test object itself. Once the MLP is trained, CoIL generates new measurements that can be used within a majority of image reconstruction methods. We validate CoIL on sparse
[2] Implicit Neural Representations with Periodic Activation Functions, Abstract: Implicitly defined, continuous, differentiable signal representations parameterized by neural networks have emerged as a powerful paradigm, offering many possible benefits over conventional representations. However, current network architectures for such implicit neural representations are incapable of modeling signals with fine detail, and fail to represent a signal's spatial and temporal derivatives, despite the fact that these are essential to many physical signals defined implicitly as the solution to partial differential equations. We propose to leverage periodic activation functions for implicit neural representations and demonstrate that these networks, dubbed sinusoidal representation networks or Sirens, are ideally suited for representing complex natural signals and their derivatives. We analyze Sir
[3] Learning Implicit Fields for Generative Shape Modeling, Abstract: We advocate the use of implicit fields for learning generative models of shapes and introduce an implicit field decoder, called IM-NET, for shape generation, aimed at improving the visual quality of the generated shapes. An implicit field assigns a value to each point in 3D space, so that a shape can be extracted as an iso-surface. IM-NET is trained to perform this assignment by means of a binary classifier. Specifically, it takes a point coordinate, along with a feature vector encoding a shape, and outputs a value which indicates whether the point is outside the shape or not. By replacing conventional decoders by our implicit decoder for representation
[4] Local Deep Implicit Functions for 3D Shape, Abstract: The goal of this project is to learn a 3D shape representation that enables accurate surface reconstruction, compact storage, efficient computation, consistency for similar shapes, generalization across diverse shape categories, and inference from depth camera observations. Towards this end, we introduce Local Deep Implicit Functions (LDIF), a 3D shape representation that decomposes space into a structured set of learned implicit functions. We provide networks that infer the space decomposition and local deep implicit functions from a 3D mesh or posed depth image. During experiments, we find that it provides
…
[20] Efficient Full-Matrix Adaptive Regularization, Abstract: Adaptive regularization methods pre-multiply a descent direction by a preconditioning matrix. Due to the large number of parameters of machine learning problems, full-matrix preconditioning methods are prohibitively expensive. We show how to modify full-matrix adaptive regularization in order to make it practical and effective. We also provide a novel theoretical analysis for adaptive regularization in non-convex optimization settings. The core of our algorithm, termed GGT, consists of the efficient computation of the inverse square root of a low-rank matrix. Our preliminary experiments show improved iteration-wise convergence rates across synthetic tasks and standard deep learning benchmarks, and that
Related work:

Figure 6: Example of truncated input text for Llama2.

LED (8192 token)

Target paper:
Title: Curvature-Aware Training for Coordinate Networks
Abstract: Coordinate networks are widely used in computer vision due to their ability to represent signals as compressed, continuous entities. However, training these networks with first-order optimizers can be slow, hindering their use in real-time applications. Recent works have opted for shallow voxel-based representations to achieve faster training, but this sacrifices memory efficiency. This work proposes a solution that leverages second-order optimization methods to significantly reduce training times for coordinate networks while maintaining their compressibility. Experiments demonstrate the effectiveness of this approach on various signal modalities, such as audio, images, videos, shape and neural radiance fields (NeRF).

[1] CoIL: Coordinate-based Internal Learning for Imaging Inverse Problems, Abstract: We propose Coordinate-based Internal Learning (CoIL) as a new deep-learning (DL) methodology for the continuous representation of measurements. Unlike traditional DL methods that learn a mapping from the measurements to the desired image, CoIL trains a multilayer perceptron (MLP) to encode the complete measurement field by mapping the coordinates of the measurements to their responses. CoIL is a self-supervised method that requires no training examples besides the measurements of the test object itself. Once the MLP is trained, CoIL generates new measurements that can be used within a majority of image reconstruction methods. We validate CoIL on sparse-view computed tomography using several widely-used reconstruction methods, including purely model-based methods and those based on DL. Our results demonstrate the ability of CoIL to consistently improve the performance of all the considered methods by providing high-fidelity measurement fields.
[2] Implicit Neural Representations with Periodic Activation Functions, Abstract: Implicitly defined, continuous, differentiable signal representations parameterized by neural networks have emerged as a powerful paradigm, offering many possible benefits over conventional representations. However, current network architectures for such implicit neural representations are incapable of modeling signals with fine detail, and fail to represent a signal's spatial and temporal derivatives, despite the fact that these are essential to many physical signals defined implicitly as the solution to partial differential equations. We propose to leverage periodic activation functions for implicit neural representations and demonstrate that these networks, dubbed sinusoidal representation networks or Sirens, are ideally suited for representing complex natural signals and their derivatives. We analyze Siren activation statistics to propose a principled initialization scheme and demonstrate the representation of images, wavefields, video, sound, and their derivatives. Further, we show how Sirens can be leveraged to solve challenging boundary value problems, such as particular Eikonal equations (yielding signed distance functions), the Poisson equation, and the Helmholtz and wave equations. Lastly, we combine Sirens with hypernetworks to learn priors over the space of Siren functions.
[3] Learning Implicit Fields for Generative Shape Modeling, Abstract: We advocate the use of implicit fields for learning generative models of shapes and introduce an implicit field decoder, called IM-NET, for shape generation, aimed at improving the visual quality of the generated shapes. An implicit field assigns a value to each point in 3D space, so that a shape can be extracted as an iso-surface. IM-NET is trained to perform this assignment by means of a binary classifier. Specifically, it takes a point coordinate, along with a feature vector encoding a shape, and outputs a value which indicates whether the point is outside the shape or not. By replacing conventional decoders by our implicit decoder for representation learning (via IM-AE) and shape generation (via IM-GAN), we demonstrate superior results for tasks such as generative shape modeling, interpolation, and single-view 3D reconstruction, particularly in terms of visual quality. Code and supplementary material are available at https://github.com/czq142857/implicit-decoder.
…
[20] Efficient Full-Matrix Adaptive Regularization, Abstract: Adaptive regularization methods pre-multiply a descent direction by a preconditioning matrix. Due to the large number of parameters of machine learning problems, full-matrix preconditioning methods are prohibitively expensive. We show how to modify full-matrix adaptive regularization in order to make it practical and effective. We also provide a novel theoretical analysis for adaptive regularization in …

Figure 7: Example of truncated input text of LED.

| Method | Dataset | Summarization | | | Structure | | | | Other |
|---|---|---|---|---|---|---|---|---|---|
| | | R-1 | R-2 | R-L | F1 | ARI | ARI' | $\mathrm{MAE}_p$ | $r_{MC}$ |
| BART | A | 0.461 | 0.131 | 0.327 | 0.664 | 0.499 | 0.482 | 1.811 | 2.79 % |
| | C | 0.425 | 0.118 | 0.310 | 0.625 | 0.396 | 0.384 | 1.963 | 5.8% |
| PEGASUS | A | 0.177 | 0.053 | 0.140 | 0.402 | 0.071 | 0.044 | 3.486 | 5.99% |
| | C | 0.181 | 0.050 | 0.135 | 0.312 | 0.121 | 0.057 | 3.364 | 12.0% |
| LED | A | 0.409 | 0.098 | 0.281 | 0.123 | 0.113 | 0.113 | 2.145 | 95.0% |
| | C | 0.415 | 0.113 | 0.285 | 0.561 | 0.261 | 0.246 | 2.275 | 5.4 % |
| Llama2-7B | A | 0.376 | 0.075 | 0.268 | 0.090 | 0.090 | 0.090 | 1.623 | 95.21% |
| | C | 0.426 | 0.089 | 0.292 | **0.700** | 0.483 | 0.469 | 1.668 | 16.8% |

Table 10: Fine-tuning result with STRoGeNS-arxiv22 and STRoGeNS-conf22. A: indicates STRoGeNS-arxiv22, C: indicates STRoGeNS-conf22

| Conferences | Summarization | | | Structure | | | | Other |
|---|---|---|---|---|---|---|---|---|
| | R-1 | R-2 | R-L | F1 | ARI | ARI' | $\mathrm{MAE}_p$ | $r_{MC}$ |
| ACL | 0.412 | 0.073 | 0.277 | 0.722 | 0.484 | 0.464 | 1.354 | 12.38 % |
| CVPR | 0.430 | 0.093 | 0.296 | 0.698 | 0.490 | 0.478 | 1.718 | 17.05 % |
| ICCV | 0.430 | 0.094 | 0.297 | 0.694 | 0.482 | 0.469 | 1.709 | 17.69 % |
| ICML | 0.413 | 0.075 | 0.279 | 0.707 | 0.465 | 0.443 | 1.673 | 17.64 % |

Table 11: Performance of Llama2 on each conference.

| Method | unigrams | bigrams | trigrams | 4-grams |
|---|---|---|---|---|
| S2ROC | 27.1 % | 73.8 % | 91.7 % | 98.2 % |
| Delve | 29.4 % | 78.1 % | 93.5 % | 98.3 % |
| TAS2 | 26.9% | 77.5% | 93.6% | 97.5% |
| TAD | 27.0 % | 77.5 % | 93.8 % | 97.5 % |
| Bigsurvey | 16.3 % | 62.1 % | 88.0 % | 95.7 % |
| SciReviewGen | 14.9 % | 59.3 % | 85.7 % | 94.0 % |
| STRoGeNS-arxiv22 | 17.4 % | 62.0 % | 86.2 % | 95.4 % |
| STRoGeNS-conf22 | 16.0% | 59.5% | 84.6% | 93.9% |
| STRoGeNS-conf23 | 14.3% | 57.1% | 84.3% | 94.4% |

Table 12: Novel $n$-grams of target sentence.

| Input | Gold standard |
|---|---|
| **Target paper:**<br>**Title:** Space Engage: Collaborative Space Supervision for Contrastive-based Semi-Supervised Semantic Segmentation **Abstract:** Semi-Supervised Semantic Segmentation (S4) aims to train a segmentation model with limited labeled images and a substantial volume of unlabeled images. To improve the robustness of representations, powerful methods introduce a pixel-wise contrastive learning approach in latent space (i.e., representation space) that aggregates the representations to their prototypes in a fully supervised manner. However, previous contrastive-based S4 methods merely rely on the supervision from the model's output (logits) in logit space during unlabeled training. In contrast, we utilize the outputs in both logit space and representation space to obtain supervision in a collaborative way. The supervision from two spaces plays two roles: 1) reduces the risk of over-fitting to incorrect semantic information in logits with the help of representations; 2) enhances the knowledge exchange between the two spaces. Furthermore, unlike previous approaches, we use the similarity between representations and prototypes as a new indicator to tilt training those under-performing representations and achieve a more efficient contrastive learning process. Results on two public benchmarks demonstrate the competitive performance of our method compared with state-of-the-art methods.<br><br>[1] Adversarial Learning for Semi-supervised Semantic Segmentation, Abstract: We propose a method for semi-supervised semantic segmentation using an adversarial network. While most existing discriminators are trained to classify input images as real or fake on the image level, we design a discriminator in a fully convolutional manner to differentiate the predicted probability maps from the ground truth segmentation distribution with the consideration of the spatial resolution. We show that the proposed discriminator can be used to improve semantic segmentation accuracy by coupling the adversarial loss with the standard cross entropy loss of the proposed model. In addition, the fully convolutional discriminator enables semi-supervised learning through discovering the trustworthy regions in predicted results of unlabeled images, thereby providing additional supervisory signals. In contrast to existing methods that utilize weakly-labeled images, our method leverages unlabeled images to enhance the segmentation model. Experimental results on the PASCAL VOC 2012 and Cityscapes datasets demonstrate the effectiveness of the proposed algorithm.<br>…<br>'[25] A Simple Framework for Contrastive Learning of Visual Representations, Abstract: This paper presents SimCLR: a simple framework for contrastive learning of visual representations. We simplify recently proposed contrastive self-supervised learning algorithms without requiring specialized architectures or a … accuracy, which is a 7% relative improvement over previous state-of-the-art, matching the performance of a supervised ResNet-50. When fine-tuned on only 1% | The aim of S4 is to train a segmentation model with the semi-supervised setting (_i.e._, a few labeled images and a large number of unlabeled images) to classify each pixel in an entire image. The critical issue of S4 is how to leverage unlabeled images to train the model. Some methods [1][2][3][4] based on GANs [5], adversarial training [6], and consistency regularization paradigm [7][8][9][10]. Meanwhile, self-training [11][12][13][14][15] is also a striking paradigm, which always generates pseudo-labels from model and retrains the model with the combined supervision of human annotations and pseudo-labels. One essential issue of self-training is the accuracy of pseudo-labels. Some methods [16][17][18][19] try to polish pseudo-labels and provide reliable guidance. Some methods [20][21][22][23] focus on the class-imbalance problems in the dataset and try to alleviate the negative effect from class-biased pseudo-labels generated by the model pre-trained on imbalanced labeled images. We build our framework based on the self-training and additionally explore semantic information among different images.<br><br>Pixel-wise contrastive learning explores semantic relations not only in the individual image but also among different images. Different from instance-wise contrastive learning [24][25][26], pixel-wise contrastive learning [27][28][29][30] project each pixel to the representation in representation space with the cooperation of encoder and representation head. Representations are then aggregated in their prototypes and are separated from each other in different classes. In semi-supervised settings, most methods [31][32][33][34] use pseudo-labels based on logits to provide semantic information contrastive learning process during training on unlabeled images. Meanwhile, the confidence of logit is used as an indicator to involve the contrastive learning process, _e.g._, [32] uses the hard representations whose corresponding logit confidence is lower than a threshold to contrast for effective training. As opposed to the above methods, we use collaborative space supervision for contrastive learning on unlabeled images and use a new indicator to involve the contrastive learning progress.<br><br>Prototype-based learning has been widely studied in few-shot learning [35][36][37] and unsupervised domain adaption [38][39][40][41][42]. Recently, it is restudied in semantic segmentation as known as a non-parametric prototype-based classifier [43]. Concretely, the classes in the dataset are presented by a set of non-learnable prototypes, and the dense semantic predictions are thus achieved by assigning the output features to its most similar prototype. Under semi-supervised settings, some methods maintain the consistency between predictions from a linear predictor and a prototype-based predictor. The two predictors are followed by the encoder and project the features to logit space and representation space, respectively. In this work, we combine the semantic information in the logit and representation spaces to provide supervision in a collaborative way during semi-supervised learning. |
| **LLAMA2-7B** | **BART** |
| **Semi-Supervised Semantic Segmentation.** Semi-supervised semantic segmentation (S4) aims to leverage abundant unlabeled data to improve the segmentation performance of limited labeled images [1][2][3][4][5][6]. The basic workflow of S4 models is first to predict the probabilities for the unlabeled images (assigning pseudo labels) and then train the model with these pseudo labels and real labeled images [7][8][9][10][11]. Although robust models trained with S4 have achieved outstanding performance, it still struggles in real-world applications due to the insufficient supervision from the labeled data [12][13][14][15][16][17][18]. In recent years, self-training (ST) based methods have emerged as a popular approach and has achieved significant improvements in segmentation accuracy and temporal stability compared with other traditional S4 methods [13][12][19][20][21][22][23].<br><br>However, previous self-training methods mainly rely on the fully-supervised model outputs in logits space to improve the robustness of feature representations. In contrast, contrastive learning has been widely used in unsupervised visual representation learning to disentangle visual features by utilizing strong supervision [24][25][26][27]. Recently, contrastive learning has been introduced to semi-supervised learning to improve the representations of images and their associated pseudo labels [28][29][30][31][32].<br><br>Most of these contrastive-based methods introduce the pseudo labels to form an augmented input dataset to supplement the limited labeled data and train the segmentation model with this augmented dataset. An Augmented Learning-based Pixel-wise Contrastive Framework (ALP) introduces a contrastive loss and a representation loss to improve the robustness of feature representation, but it is designed to work with a fully-supervised image labeling scheme in the segmentation pipeline (i.e., images are labelled before training). Also, the inputs used in ALP are obtained with the ground truth labels. S4[33] simply selects highly confident pixels as pseudo labels and updates the model with the augmented dataset, thus does not have similar supervision as ALP. In contrast, C3-SS [14] trains a segmentation model with pseudo labels in logit space and the ground-truth labels in representation space. The features extracted from the logit-space segmentation model are fed into a contrastive loss to enhance their robustness. We also use the features in logit space to produce a contrastive loss, but unlike previous contrastive-based S4 methods, we propose to enhance the similarity between the representations and prototypes with an extra representation-to-prototype contrastive loss. Furthermore, unlike C3-SS which only uses the network outputs in | **Semi-Supervised Semantic Segmentation (SSL)** aims to learn a segmentation model with limited labeled data and a substantial volume of unlabeled data. Adversarial-based methods [1][2][3][4] utilize GANs [5] to generate pseudo-labels for unlabelled data. VAT [6] and CCT [7] use adversarial training to encourage the predictions to be consistent with the ground truth. UCC [8] and CPS [9] use uncertainty estimation to enhance pseudo-label generation. Transformer-CNN Cohort [10] introduces a self-training framework that combines a teacher model and a student model. Self-training methods [11][12][13][14][15][16][17][18][19][20][21][22][23] are also widely used in S4. FixMatch [18] combines self-supervised learning and consistency regularization. ST++ [19] improves FixMatch by applying multi-level perturbations. CReST [21] uses class-rebalancing to improve the quality of pseudo labels.<br><br>Contrastive-based Learning [24][25][26] has achieved great success in unsupervised visual representation learning. Recently, contrastive learning has also been introduced to SSL. C3-SemiSeg [14] uses a contrastive loss in the pixel-level. Pixel-wise contrastive methods [27][28][29][30][31][32][33][34] also show promising results. PiPa [29] and PiPaV2 [27] perform pixel-wise and feature-level contrastive losses. PCC [30] and PCC-v2 [31] use a consistency loss between global and local features. UCT [33] uses unreliable predictions from the teacher model to supervise the student model, which is similar to Mean-Teacher [12].<br><br>Prototype-based learning [35][36][37][38][39][40][41][42] has been applied to many tasks, such as few-shot learning and domain adaptation. PCL [34] is the first work to use prototypes to boost the performance of SSL. However, PCL only uses the supervision from the output in logit space. In contrast, we utilize the outputs in both logit and representation spaces to supervize each other.<br><br>proposes a contrast-based method to learn discriminative features for image classification. SimCLR [25] proposes a simple framework that performs instance discrimination in the latent space. Inspired by these works, we propose a pixel-wised contrastive method for S4, which differs from previous works in two aspects. First, we use the similarity between representations and prototypes as a new indicator to tilt training those under-performing representations. Second, we enhance the knowledge transfer between the two spaces by introducing a collaborative learning strategy.<br><br>introduces a contrast loss to learn transferable features in a fully supervised manner. [43] extends this idea to S4 and proposes a dual-path self-distillation method to improve |

Figure 8: Example of generated related work section 1. Both BART and Llama2-7B results are organized research topics with multiple paragraphs and contain novelty statements. Red text indicate sentence of novelty statement.

| Input | Gold standard |
|---|---|
| **Target paper:**<br>**Title:** Don't Generate, Discriminate: A Proposal for Grounding Language Models to Real-World Environments<br>**Abstract:** A key missing capacity of current language models (LMs) is grounding to real-world environments. Most existing work for grounded language understanding uses LMs to directly generate plans that can be executed in the environment to achieve the desired effects. It thereby casts the burden of ensuring grammaticality, faithfulness, and controllability all on the LMs. We propose Pangu, a generic framework for grounded language understanding that capitalizes on the discriminative ability of LMs instead of their generative ability. Pangu consists of a symbolic agent and a neural LM working in a concerted fashion: The agent explores the environment to incrementally construct valid plans, and the LM evaluates the plausibility of the candidate plans to guide the search process. A case study on the challenging problem of knowledge base question answering (KBQA), which features a massive environment, demonstrates the remarkable effectiveness and flexibility of Pangu: A BERT-base LM is sufficient for setting a new record on standard KBQA datasets, and larger LMs further bring substantial gains. Pangu also enables, for the first time, effective few-shot in-context learning for KBQA with large LMs such as Codex.1Footnote 1: The Pangu library: OSU-NLP-Group/Pangu.<br><br>[1] A Comprehensive Exploration on WikiSQL with Table-Aware Word Contextualization, Abstract: We present SQLova, the first Natural-language-to-SQL (NL2SQL) model to achieve human performance in WikiSQL dataset. We revisit and discuss diverse popular methods in NL2SQL literature, take a full advantage of BERT {Devlin et al., 2018) through an effective table contextualization method, and coherently combine them, outperforming the previous state of the art by 8.2% and 2.5% in logical form and execution accuracy, respectively. We particularly note that BERT with a seq2seq decoder leads to a poor performance in the task, indicating the importance of a careful design when using such large pretrained models. We also provide a comprehensive analysis on the dataset and our model, which can be helpful for designing future NL2SQL datsets and models. We especially show that our model's performance is near the upper bound in WikiSQL, where we observe that a large portion of the evaluation errors are due to wrong annotations, and our model is already exceeding human performance by 1.3% in execution accuracy.<br>…<br>[23] SmBoP: Semi-autoregressive Bottom-up Semantic Parsing, Abstract: The de-facto standard decoding method for semantic parsing in recent years has been to autoregressively decode the abstract syntax tree of the target program using a top-down depth-first traversal. In this work, we propose an alternative approach: a Semi-autoregressive Bottom-up Parser (SmBoP) that constructs at decoding step t the top-K sub-trees of height ≤ t. Our parser enjoys several benefits compared to top-down …semantically-vacuous partial trees. We apply SmBoP on Spider, a challenging zero-shot semantic parsing benchmark, and show that SmBoP leads to a 2.2x speed-up in decoding time and a ~5x speed-up in training time, compared to a semantic parser that uses autoregressive decoding. SmBoP obtains 71.1 denotation accuracy on Spider, establishing a new state-of-the-art, and 69.5 exact match, comparable to the 69.6 exact match of the autoregressive RAT-SQL+Grappa. | The Seq2Seq framework ([5]; [4]) has been the _de facto_ choice for grounded language understanding, where the LM directly generates a plan given an input utterance. However, the lack of grounding during pretraining makes generating valid plans from LMs challenging. Recent studies endeavor to alleviate this issue via _input augmentation_ or _constrained decoding_. For input augmentation, the environment (or some relevant portion of it) is fed to the LM's encoder together with the utterance ([1]; [6]; [7]). Such methods rely on the LM to understand the interplay between the language requests and the environment and correctly factor that into plan generation. They therefore require substantial training data to learn and also provide no guarantee for grammaticality or faithfulness. In contrast, constrained decoding methods regulate the decoder's behavior to guarantee grammaticality ([8]; [3]) or even faithfulness ([9]; [2]). However, such uses still cast the burden of generating valid plans on the LM itself; controlling the generation process of an LM can be difficult and specific to each planning language and/or environment. In our proposal, the LM is only used to discriminate valid plans proposed by an agent through a controllable search process. More detailed comparison is presented in SS5.3.<br><br>Large language models (LLMs) (; [17]) have demonstrated strong few-shot learning capabilities in various tasks, from writing programs to query structured and unstructured data ([16]; [13]; [10]), interacting with online websites ([11]; [14]), to generating procedural plans and guiding embodied agents in virtual environments ([15]; [18]; [12]; ). Most existing work still capitalizes on the generative ability of LLMs. A common strategy to encourage an LLM to produce valid plans is to directly _describe_ the environment in the LLM's context (_i.e._, input augmentation), which is difficult for complex environments like KBs. A concurrent work of ours () asks the LLM to directly generate a proxy plan from the input question without the environment description, which is then used to retrieve a valid plan from a set of candidate plans. However, this design is tailored specifically to the KB query language and is limited to generating plans with at most two hops due to the combinatorial explosion in their candidate enumeration. In contrast, Pangu shields the LLM from the complexity of the environment and lets the LLM focus on evaluating the plausibility of candidate plans proposed by an agent. One interesting related work is [18], where an LLM is used to score atomic action (skill) proposals, which are guaranteed to conform to affordance constraints, from an embodied agent. Pangu shares a similar spirit of using LMs for discrimination, but we support more complex plans through a search process in the environment guided by an LM.<br><br>Our instantiation of Pangu on KBQA is closely connected to bottom-up semantic parsing, particularly SmBoP ([23]), a text-to-SQL model that iteratively constructs a complex plan from a set of subplans. Pangu similarly constructs a complex plan incrementally from smaller subplans, but it makes the following main departures. First, SmBoP requires all ingredients (_i.e._, column headers, table names, and DB values) at the beginning of parsing. This assumption does not generally hold for more complex or partially observable environments, where ingredients need to be discovered through search. In our method, only topic entities are needed as the initial plan, which can be readily obtained using an entity linker ([21]). Second, our scoring function is based on a straightforward application of LMs, while SmBoP uses a more intricate architecture with extra parameters. Also related is an array of earlier KBQA methods that adopt an enumerate-and-rank approach ([20]; [19]; [22]). Because they try to enumerate all candidate plans up front, the maximum plan complexity is bound to be small. Our adaptive search process allows for flexible construction of more complex plans. |
| LLAMA2-7B | BART |
| **Language Understanding with Language Models.** Large pre-trained language models (LMs) have emerged as powerful tools for a variety of tasks in natural language processing ([5]; [4]). In the past years, great efforts have been devoted to understanding the knowledge encoded in LMs ([6]; ; [1]), bridging LMs and traditional program synthesis methods ([16]; ; [12]; [15]), and transferring LMs to different tasks, such as text-to-SQL () and text-to-text (; ; [7]; [13]; [9]; [14]; [11]; [3]; [8]; [2]; [10]; [17]; [18]).<br><br>**Knowledge Base Question Answering.** Recent research on KBQA aims to ground LMs on knowledge bases to enhance the reasoning and comprehension abilities ([22]; [23]; [20]; ; [21]; [2]). Specifically, several works train general-purpose LMs on a large amount of KBQA data to improve the reasoning ability ([22]; [23]), and others use a hyper-parameter-tuning strategy to deal with the out-of-distribution (OOD) KBQA problem ([24]; [19]; ). Differently, the proposed Pangu directly implements grounding. | **Knowledge Base Question Answering.** Knowledge base question answering (KBQA) is a long-standing problem in natural language processing (NLP) [19]; ; [3]; [1]; [7]; [2]; [6]. Most existing work on KBQA is based on the Seq2Seq framework [5]; [4], which encodes the question and the knowledge base into a joint embedding space, and then decodes the answer to the question using a decoder that takes the embedding as input.<br><br>**Grounded Language Understanding.** Recently, there has been a surge of interest in grounding language models to symbolic languages, such as code [17]; [10]; [9]; [8], HTML [11]; [14], and SQL [13]; ; ; [6]; ;. Most of the existing work for grounded language understanding uses LMs to directly generate plans that can be executed in the environment to achieve the desired effects. This casts the burden of ensuring grammaticality, faithfulness, and controllability all on the LMs. In contrast, Pangu leverages the discriminative ability of LMs instead of their generative ability, and thus enables, for the first time, grounding to real-world environments. Our work is also related to [15]; [12]; [18]; [16], which also use LMs for task planning. However, unlike Pangu, these methods do not explicitly consider the plausibility of the generated plans, and hence cannot guarantee the correctness of the plans. In addition, they do not scale well to large LMs, which are required for our applications.<br><br>**Semantic Parsing.** Semantic parsing is the task of converting natural language questions into executable programs, which can then be executed by a symbolic agent to obtain the answer [20]; [23]; [21]; [22]. In particular, [20] propose a staged query graph generation (SQG) framework, which first generates a query graph from the question, and subsequently uses it to answer the question. [21] propose an end-to-end entity linking framework that first generates an entity linking graph and then links the entities in the graph to the answer. [23] propose to use a semi-progressive bottom-up semantic parsing framework to first generate a set of candidate answers and then select the best answer from the top-k answers. In this work, we propose to leverage a discriminator to guide the search process of the symbolic agent, so that the generated programs can be more likely to be valid.<br><br>propose a framework for symbolic question answering, which is similar to our work in spirit, but differs in several important ways. First, they only consider the question as input, and do not consider the context in which the question is expressed. Second, their framework only considers the question in the context of a single question, while our framework considers the context from multiple questions. Third, their work only focuses on SQL, whereas our work focuses on the more challenging task of knowledge graph question answering. Fourth, our framework can scale to larger LMs and |

Figure 9: Example of generated related work section 2. BART tends to contain novelty statements.

## F Datasheets for datasets

### Motivation

**For what purpose was the dataset created?** Was there a specific task in mind? Was there a specific gap that needed to be filled? Please provide a description.

The dataset was created to enable research to generate structured related work with novelty statement: given target article information and cited articles information, generate related work section with paragraph whether paragraphs are written based topics determined from the relations between current work and previous work and novelty statement.

**Who created this dataset (e.g., which team, research group) and on behalf of which entity (e.g., company, institution, organization)?**

OMRON SINIC X Corp.

**Who funded the creation of the dataset?** If there is an associated grant, please provide the name of the grantor and the grant name and number.

See acknowledgement.

### Composition

**What do the instances that comprise the dataset represent (e.g., documents, photos, people, countries)?** Are there multiple types of instances (e.g., movies, users, and ratings; people and interactions between them; nodes and edges)? Please provide a description.

The instances are related work sections extracted from scientific articles, together with the title and abstract of the target article and cited paper information (*i.e.,* title, abstract).

**How many instances are there in total (of each type, if appropriate)?**

See Table 1, 2, 3, 4.

**Does the dataset contain all possible instances or is it a sample (not necessarily random) of instances from a larger set?** If the dataset is a sample, then what is the larger set? Is the sample representative of the larger set (e.g., geographic coverage)? If so, please describe how this representativeness was validated/verified. If it is not representative of the larger set, please describe why not (e.g., to cover a more diverse range of instances, because instances were withheld or unavailable).

Our dataset relies on open-source top conference proceedings. Therefore, we can collect more instances by parsing the proceedings.

**What data does each instance consist of? "Raw" data (e.g., unprocessed text or images) or features?** In either case, please provide a description.

Each instance consists of the text associated with the articles. We only used articles that could correctly extract the title, abstract, related work, and the list of cited articles and removed articles with a related work paragraph of 1 and 8 or more paragraphs or fewer than five cited articles. All citation identifiers such as Author and year are replaced [#i].

**Is there a label or target associated with each instance?** If so, please provide a description.

Randomly sampled 100 samples were annotated for novelty statement sentences for novelty evaluation.

**Is any information missing from individual instances?** If so, please provide a description, explaining why this information is missing (e.g., because it was unavailable). This does not include intentionally removed information, but might include, e.g., redacted text.

As shown in Table 5, some samples missing abstract of cited article information cause retrieval errors of semantic scholar API. In the conference datasets, some title of cited articles are missing due to parse errors of NOUGAT.

**Are relationships between individual instances made explicit (e.g., users' movie ratings, social network links)?** If so, please describe how these relationships are made explicit.

None explicitly. Some instances are cited in other instances.

**Are there recommended data splits (e.g., training, development/validation, testing)?** If so, please provide a description of these splits, explaining the rationale behind them.

We recommended using STRoGeNS-arXiv22, STRoGeNS-conf22 as training and STRoGeNS-arXiv23 as test to avoid data leak. The dataset is separated by published years of articles and the split setting reflects the actual usage of related work generation.

**Are there any errors, sources of noise, or redundancies in the dataset?** If so, please provide a description.

The datasets collected from the top conferences may contain parse errors of NOUGAT.

**Is the dataset self-contained, or does it link to or otherwise rely on external resources (e.g., websites, tweets, other datasets)?** If it links to or relies on external resources, a) are there guarantees that they will exist, and remain constant, over time; b) are there official archival versions of the complete dataset (i.e., including the external resources as they existed at the time the dataset was created); c) are there any restrictions (e.g., licenses, fees) associated with any of the external resources that might apply to a future user? Please provide descriptions of all external resources and any restrictions associated with them, as well as links or other access points, as appropriate.

STRoGeNS-arxiv2022 contains OpenAlex links for each citation and license link.

**Does the dataset identify any subpopulations (e.g., by age, gender)?** If so, please describe how these subpopulations are identified and provide a description of their respective distributions within the dataset.

See Table 2, 3, and 4.

## Collection Process

**How was the data associated with each instance acquired?** Was the data directly observable (e.g., raw text, movie ratings), reported by subjects (e.g., survey responses), or indirectly inferred/derived from other data (e.g., part-of-speech tags, model-based guesses for age or language)? If data was reported by subjects or indirectly inferred/derived from other data, was the data validated/verified? If so, please describe how.

The data is observable as raw text.

**What mechanisms or procedures were used to collect the data (e.g., hardware apparatus or sensor, manual human curation, software program, software API)?** How were these mechanisms or procedures validated?

To collect the conference PDFs, we used Beautiful Soup and parsed by NOUGAT. Then, abstracts of citations are collected by Semantic Scholar API.

**Who was involved in the data collection process (e.g., students, crowdworkers, contractors) and how were they compensated (e.g., how much were crowdworkers paid)?**

Two researchers collected the dataset.

**Over what timeframe was the data collected? Does this timeframe match the creation timeframe of the data associated with the instances (e.g., recent crawl of old news articles)?** If not, please describe the timeframe in which the data associated with the instances was created.

Conference datasets are collected from conference proceedings from 2017 to 2023. unarXiv2022 is a collected articles uploader until 2022.

## Preprocessing/cleaning/labeling

**Was any preprocessing/cleaning/labeling of the data done (e.g., discretization or bucketing, tokenization, part-of-speech tagging, SIFT feature extraction, removal of instances, processing of missing values)?** If so, please provide a description. If not, you may skip the remainder of the questions in this section.

The samples that had one paragraph, more than eight paragraphs, and less than five citations were removed. Regarding conference datasets, we only use related work sections that collected the title, the abstract, the related work section, and the references of the target paper.

**Was the "raw" data saved in addition to the preprocessed/cleaned/labeled data (e.g., to support unanticipated future uses)?** If so, please provide a link or other access point to the "raw" data.

Yes.

## Distribution

**Will the dataset be distributed to third parties outside of the entity (e.g., company, institution, organization) on behalf of which the dataset was created?** If so, please provide a description.

STRoGeNS-arXiv22 is constructed based on each license of arXiv papers. Therefore, please follow the license of arXiv paper.

**How will the dataset be distributed (e.g., tarball on website, API, GitHub)** Does the dataset have a digital object identifier (DOI)?

We will publish processing codes to Github and dataset by the website as long as the license allows.

**When will the dataset be distributed?**

The paper is accepted.

**Will the dataset be distributed under a copyright or other intellectual property (IP) license, and/or under applicable terms of use (ToU)?** If so, please describe

this license and/or ToU, and provide a link or other access point to, or otherwise reproduce, any relevant licensing terms or ToU, as well as any fees associated with these restrictions.

ACL, EMNLP, NAACL, and ICLR are follows licensed under the Creative Commons Attribution-NonCommercial-ShareAlike 3.0 International License. CVPR, ICCV, ECCV, ICML, and arXiv follow all rights therein are retained by authors or by other copyright holders.