VTechAGP: An Academic-to-General-Audience Text Paraphrase Dataset and Benchmark Models

Ming Cheng[1*], **Jiaying Gong**[1*], **Chenhan Yuan**[2],
William A. Ingram[1], **Edward Fox**[1], **Hoda Eldardiry**[1]
[1]*Virginia Tech,* [2]*University of Manchester*
{ming98,gjiaying,waingram,fox,hdardiry}@vt.edu, *chenhan.yuan@manchester.ac.uk*

Abstract

Existing text simplification or paraphrase datasets mainly focus on sentence-level text generation in a general domain. These datasets are typically developed without using domain knowledge. In this paper, we release a novel dataset, **VTechAGP**, which is the first *academic-to-general-audience* text paraphrase dataset consisting of document-level thesis and dissertation academic and general-audience abstract pairs from 8 colleges authored over 25 years. We also propose a novel *dynamic soft prompt* generative language model, **DSPT5**, for the academic-to-general-audience text paraphrasing task. For training, we leverage a contrastive-generative loss function to learn the keyword vectors in the dynamic prompt. For inference, we adopt a crowd-sampling decoding strategy at both semantic and structural levels to further select the best output candidate. We evaluate **DSPT5** and various state-of-the-art large language models (LLMs) from multiple perspectives. Results demonstrate that the SOTA LLMs do not provide satisfactory outcomes, while the lightweight **DSPT5** can achieve competitive results. To the best of our knowledge, we are the first to build a benchmark dataset and solutions for academic-to-general-audience text paraphrase dataset [1].

1 Introduction

Text generation aims to produce understandable text in human language from various sources of input data. Among them, text-to-text generation remains an important and challenging task with extensive applications such as language translation (Ranathunga et al., 2023; Dabre et al., 2020), paraphrase generation (Singh and Josan, 2022), text simplification (Martin et al., 2023), etc.

Existing text simplification and paraphrase generation datasets (shown in Table 5 in the Appendix) mainly focus on sentence-level translation. The recent paragraph-level dataset WikiAuto (Jiang et al., 2020), derived from WikiLarge (Zhang and Lapata, 2017) and Newsela (Xu et al., 2015), still suffers from a lack of domain diversity and specification, which are sourced from Wikipedia and news articles. Besides, the objective of these studies on the aforementioned datasets is to simplify the text for children at lower grade levels. Furthermore, existing works on lay summarization involve brief summarization limited to medicine and biological domains (Devaraj et al., 2021; Flores et al., 2023; Jiang et al., 2024; Zaman et al.). However, our proposed VTechAGP involves translation from technical language to general-audience language across a broader set of multiple domains. We aim to generate text from the domain-specific to a general level of understanding while keeping it scientifically accurate and easy to understand, in order to encourage and facilitate interdisciplinary collaboration across different research fields.

In this paper, we pioneer the research of academic-to-general-audience text generation by introducing a new benchmark dataset VTechAGP, which is derived from electronic theses and dissertations (ETDs) at Virginia Tech over twenty-five years. VTechAGP consists of document-level abstract pairs (academic abstract and general-audience abstract). VTechAGP also provides other information such as title, discipline, degree level, etc. This auxiliary information shows the potential of VTechAGP for other tasks such as topic generation, etc. In addition, the abstracts in VTechAGP come from multiple domains (colleges) and VTechAGP is labeled with each specific domain and provides the domain knowledge keywords. More details about VTechAGP are presented in Sec. 3 and Table 6.

Based on VTechAGP, we evaluate several SOTA pre-trained large language models (LLMs), such as LLaMA2 (Touvron et al., 2023), Claude2 (An-

[*]Ming cheng and Jiaying Gong contributed equally.
[1]https://github.com/waingram/VTechAGP-Dataset, Dataset: https://doi.org/10.5281/zenodo.14833932

Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers), pages 6110–6130

thropic, 2023), ChatGPT (Brown et al., 2020), etc., to establish the baseline performance. However, these SOTA pre-trained LLMs have demonstrated the following limitations: (1) Some LLMs do not provide public APIs, or the APIs are not free. Also, some LLMs (e.g., Claude2) do not provide fine-tuning, making them less adaptable to specific tasks. (2) The model size of LLMs is very large. For example, LLaMA2 has about 65 billion parameters. Fine-tuning these LLMs is resource-intensive in terms of memory and computation time. Even the inference implementation requires more memory and time. (3) The pre-trained LLMs do not show competitive performance for the academic-to-general-audience text paraphrasing task on VTechAGP in Sec. 5.

To address the above challenges, we propose DSPT5, a dynamic soft prompt-based generative model with the crowd sampling decoding strategy during the inference stage. DSPT5 is built based on the pre-trained T5 (Raffel et al., 2020), which has only about 220 million parameters. In particular, the dynamic soft prompt template in DSPT5 can automatically adapt to different academic domains by changing the keywords extracted from the academic abstract. The prompt encoder in DSPT5 is trained to generate and fine-tune keyword vectors combined with the dynamic prompt template. To this end, we design a hybrid loss function with generative language model loss and contrastive loss to jointly learn the generated text representations as well as the ability to distinguish technical keywords from non-technical keywords. During inference, DSPT5 employs two alignment functions at both the semantic and structural levels to select the best candidate for the final generated output.

The contributions can be summarized as follows: (1) **Dataset:** We construct VTechAGP, the first academic-to-general-audience text paraphrase dataset. VTechAGP is a document-level text generation dataset with multiple technical domains. (2) **Baselines:** We implement several SOTA LLMs as benchmarks to compare the performance with our proposed model DSPT5. Experimental results show that there is still a huge room for further improvement of the existing LLMs. (3) **Approach:** We propose a lightweight model, DSPT5, which utilizes dynamic soft prompts with a hybrid loss function and a new crowd decoding strategy. Experimental results show that DSPT5 can achieve competitive results with SOTA LLMs. (4) **Evaluation:** We explore various evaluation metrics for the academic-to-general-audience text paraphrasing task on VTechAGP from different perspectives, including document-level embedding-based, word-based, and end-to-end metrics. In addition, simplicity, diversity, readability, and toxicity are also considered for the performance evaluation.

2 Related Work

2.1 Text Generation Datasets

The most commonly used datasets for text generation focus on cross-lingual machine translation, such as WMT datasets (Farhad et al., 2021; Bojar et al., 2016) for news translation, IWSLT datasets (Scarton et al., 11 2-3 2019; Lee et al., 2022) for document translation of TED talks, parallel corpus datasets (Europarl (Koehn et al., 2003), UN Parallel Corpus (Ziemski et al., 2016), OPUS (Zhang et al., 2020), and Tatoeba (Tiedemann, 2020)) for sentence-level translation. Additional text generation datasets focus on text/sentence simplification, such as datasets from news articles (Xu et al., 2015; Stodden et al., 2023), clinical reports (Luo et al., 2022a), Wikipedia (Zhu et al., 2010; Xu et al., 2016a; Zhang and Lapata, 2017; Alva-Manchego et al., 2020; Naderi et al., 2019; Aumiller and Gertz, 2022), and other language learning resources (Vajjala and Lučić, 2018a). Existing paraphrase datasets include phrasal and lexical paraphrases (Ganitkevitch et al., 2013), question pairs from Wikidata (Fader et al., 2013), and Quora (Wang et al., 2017), the same posts shared by Twitter (Lan et al., 2017), different image captions (Lin et al., 2014), and a machine translation dataset involving back translation (Huang et al., 2023). However, these datasets are limited to sentence-level translation and non-academic data with a lack of domain expert knowledge.

2.2 Text Generation Methods

With the emergence of Transformers (Vaswani et al., 2017), pre-trained language models (PLMs) – such as BERT (Devlin et al., 2019), BART (Lewis et al., 2020), ChatGPT (Brown et al., 2020), T5 (Raffel et al., 2020), Claude2 (Anthropic, 2023), and LLaMA2 (Touvron et al., 2023) – have been developed and fine-tuned for various generation tasks. For example, fine-tuning LLMs on a customized dataset (Sun et al., 2023; Yermakov et al., 2021; Dathathri et al., 2020) with prompts (Luo et al., 2023; Wan et al., 2023; Yang et al., 2023;

Table 1: Dataset statistics of VTechAGP over eight colleges. Statistics are reported in the format of academic (source) / general audience (target) documents.

College	Avg. # Sentence	Avg. Sentence Len.	MTLD	Readability Consensus
Agriculture and Life Sciences (ALS)	15.06 / 13.80	27.42 / 26.37	73.59 / 68.58	15th-16th / 14th-15th
Architecture, Arts, and Design (AAD)	10.83 / 8.99	26.67 / 26.07	68.24 / 68.76	13th-14th / 10th-11th
Engineering (ENG.)	14.03 / 11.99	26.87 / 25.66	69.93 / 66.47	13th-14th / 14th-15th
Liberal Arts and Human Sciences (LAHS)	10.12 / 8.60	30.24 / 28.46	68.02 / 67.06	16th-17th / 15th-16th
Natural Resources and Environment (NRE)	14.48 / 12.66	28.87 / 27.33	76.69 / 71.99	15th-16th / 10th-11th
Science (SCI.)	14.19 / 11.75	27.35 / 25.73	74.90 / 68.48	16th-17th / 15th-16th
Business (BUS.)	11.76 / 10.24	28.55 / 27.63	68.46 / 70.35	18th-19th / 17th-18th
Veterinary Medicine (VM)	15.61 / 13.16	26.83 / 25.92	77.77 / 70.15	16th-17th / 14th-15th
All	13.68 / 11.66	27.68 / 26.38	71.36 / 67.91	15th-16th / 13th-14th

Kew et al., 2023; Chen et al., 2023) is a popular method for text generation. Several decoding strategies (i.e, minimum Bayes decoding (Suzgun et al., 2023), beam search (Yoon and Bak, 2023; Zhang et al., 2020), probability-based sampling (Li et al., 2022; Guo et al., 2018; Xu et al., 2022), nucleus sampling (Holtzman et al., 2020), prefix-adaptive decoding (Pei et al., 2023), and contrastive loss (An et al., 2022)) are applied to select the best generated candidates. However, these studies only focus on paraphrase generation or text simplification in a general domain. Our focus is not on shortening the text, but rather on how these PLMs can paraphrase academic language into non-academic language while maintaining accuracy and simplicity.

3 Dataset Construction

3.1 Data Source and Dataset Collection

Virginia Tech has been a leader in ETDs for more than twenty-five years. It was the first university to require electronic submission of ETDs, beginning in 1997. Virginia Tech's ETDs are accessible through VTechWorks,[2] a digital repository created through a collaboration between the Graduate School and the University Libraries. In the fall of 2016, the Graduate School added a new requirement for ETDs: the inclusion of a *general audience abstract* in addition to the traditional academic abstract. The ETD submission system was updated in 2019 to include a separate field for the general audience abstract. Since then, most ETDs have included both an academic abstract and a general audience abstract, which are captured as distinct metadata fields in VTechWorks. Like many institutional repositories, VTechWorks supports the Open Archives Initiative Protocol for Metadata Harvesting (OAI-PMH) (Lagoze et al., 2002). OAI-PMH is a standard protocol for retrieving metadata records from digital repositories. It provides a framework that enables data exchange between various systems, supporting consistent metadata formats. To create the VTechAGP dataset, we used OAI-PMH to harvest metadata from VTechWorks. We specifically queried the VTechWorks OAI-PMH endpoint to retrieve metadata records for ETDs containing both academic and general audience abstracts. The OAI-PMH endpoint provided us with an XML record for each ETD. This data encompassed a range of metadata elements defined in a qualified Dublin Core schema. We identified specific fields necessary for our dataset and extracted the text content. This step involved parsing the XML structure, locating the relevant elements, and retrieving the textual data they contained. We mapped the extracted metadata to specific columns in a CSV file. Each piece of extracted metadata was mapped to a corresponding column header in the CSV file: 'identifier_uri', 'title', 'abstract', 'abstract_general', 'subject_terms', 'discipline', 'department', 'degree', 'degree_level', and 'type'. A description is given in Table 6 in the Appendix. We provide the data analysis in Sec. B.

3.2 Task Definition

We name the dataset VTechAGP and propose the academic to the general audience text paraphrase task. Let $D = \{X_n, Y_n, A_n\}_{n=1}^{N}$ denote a dataset where X_n is the source document, Y_n is the target document, A_n is the auxiliary information (subject terms, which are the keywords in ETDs provided by the author) of the source document, and N is the number of documents. Given an input sequence of words $X = [x_1, \cdots, x_S]$ with length S and auxiliary information $A = [a_1, \cdots, a_A]$ with length A, we aim to generate an output sequence of words $Y = [y_1, \cdots, y_T]$ with length T that retains the original meaning as X, but reduces the complexity

[2]Virginia Tech Electronic Theses and Dissertations: https://hdl.handle.net/10919/5534

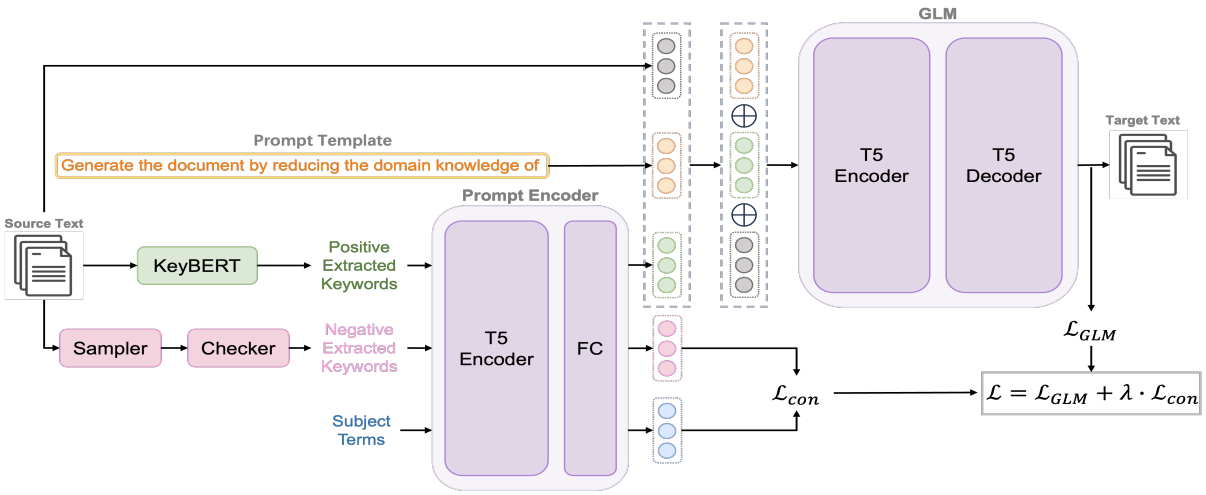

Figure 1: Dynamic soft prompt generation of our proposed model DSPT5. The framework includes a prompt encoder and a generative language model fine-tuning together with a hybrid loss. FC is a fully connected layer.

to improve readability and comprehension without domain knowledge. In this task, our goal is to find the set of model parameters that maximize $\prod_{n=1}^{N} p_{model}(Y_n|X_n)$.

4 Approach

Figure 1 shows our proposed DSPT5 with two main components: dynamic soft prompt generation with hybrid loss and crowd sampling decoding strategy.

4.1 Backbone

A generative language model prescribes the generation of a sentence as a sequence of word predictions based on context. T5 (Raffel et al., 2020) has shown high performance and few-shot abilities on various language understanding tasks, including text simplification and language translation (Brown et al., 2020). The decoding capacity of the generative language model can help to generate arbitrary content well when given appropriate prompts. Therefore, we use the pre-trained version of T5 to initialize our model DSPT5. We fine-tune T5 with dynamic prompts on the training set of VTechAGP [3]. During the inference stage, we implement the crowd sampling decoding strategy to better select word candidates for text generation.

4.2 Dynamic Soft Prompt Generation

In our proposed DSPT5, we introduce an automatic customized soft prompt generation process that includes three main steps: (1) Dynamic Prompt Generation, (2) Soft Prompt Encoder, and (3) Model Training with Hybrid Loss.

4.2.1 Dynamic Prompt Generation

To better control the output generated by T5 for our task, we added a prompt template before the source input (academic abstracts). Instead of using a static prompt template, such as "Generate another version of the provided document for general audiences", we design a dynamic prompt template that can learn to generate academic keywords to adapt to different academic domains. As opposed to adding the keywords directly in front of the prompt (Blinova et al., 2023), we designed a template (which is used in the experiments): "Generate the document by reducing the domain knowledge of " + keywords, where the keywords (auxiliary information) $K^{pos} = [k_1^{pos}, \cdots, k_N^{pos}]$ are generated by KeyBERT (Grootendorst, 2020) [4] and sorted by their importance score:

$$K^{pos} = sorted(KeyBERT(X_i)) \quad (1)$$

N is the number of keywords extracted from the source text by KeyBERT. Since KeyBERT does not support fine-tuning on custom datasets, we develop a soft prompt encoder in Sec. 4.2.2 to further fine-tune the embeddings of K^{pos} by minimizing the distance between the embeddings of K^{pos} and the embeddings of the subject terms from the auxiliary information A (golden label). Therefore, dynamic prompts can be automatically generated using the keywords extracted from KeyBERT, and the fine-tuned embeddings of such keywords can be ob-

[3] FLAN-T5 shows a worse performance based on our task

[4] https://github.com/MaartenGr/KeyBERT

tained by the soft prompt encoder in the inference stage. Details of loss are introduced in Sec. 4.2.3.

4.2.2 Soft Prompt Encoder

We propose a prompt encoder aiming to generate a soft prompt representing the hidden representation of the text to be generated. The prompt generator can be any sequence-to-sequence model. We use the T5 (Raffel et al., 2020) encoder as the backbone for the prompt encoder. We first obtain the embeddings $E_{prompt} = [e_1, \cdots, e_A]$ and $E_X = [e_1, \cdots, e_S]$ of a given sequence of tokens $P = [p_1, \cdots, p_A]$ and $X = [x_1, \cdots, x_S]$ from the partial fixed prompt and the source text, respectively. Next, we get the output representations of the prompt encoder r_i, which can be formulated as:

$$\begin{aligned} h_i^{key} &= f_\phi(A_i) \\ r_i^{key} &= ReLU(W \cdot h_i^{key} + b) \end{aligned} \tag{2}$$

where f_ϕ is the prompt (T5) encoder, h_i is the last hidden state of the T5 encoder, and A_i has the keywords of the subject terms. A fully connected layer is added after the T5 encoder, where $W \in \mathbb{R}^{n \times d}$ and $b \in \mathbb{R}^n$ are trainable. n and d are the length and dimension of the keywords. Similarly, we get the representations of the extracted keywords r_i^{pos}:

$$\begin{aligned} h_i^{pos} &= f_\phi(K^{pos}) \\ r_i^{pos} &= ReLU(W \cdot h_i^{pos} + b) \end{aligned} \tag{3}$$

where K^{pos} is from Equ. 1. We construct negative samples to cover non-academic words. As shown in Figure 1, we implement a sampler that randomly samples words from the source text, where the embeddings of the negative samples r_i^{neg}:

$$\begin{aligned} K^{neg} &= Random(X_i) \\ h_i^{neg} &= f_\phi(K^{neg}) \\ r_i^{neg} &= ReLU(W \cdot h_i^{neg} + b) \end{aligned} \tag{4}$$

where X_i is the input source document. The checker in Figure 1 ensures that $K^{neg} \cap K^{pos} = \varnothing$.

4.2.3 Model Training with Hybrid Loss

Given the representations of the extracted soft keyword prompts r_i^{pos}, the embeddings of the prompt template E_{prompt}, and the source input E_X, the output target text generated by our proposed model:

$$\begin{aligned} P_\theta(\cdot|x) &= GLM(\{E_{prompt} \oplus r^{pos} \oplus E_X\}) \\ \hat{y} &\sim P_\theta(\cdot|x) \end{aligned} \tag{5}$$

where GLM is the T5 backbone, θ is the model parameter, $\oplus$ denotes concatenation, and $\hat{y}$ is the generated general audience document. DSPT5 is trained to fit the mapping from academic abstracts to general audience abstracts. Formally, let the dataset be D with size N. DSPT5 aims to maximize the standard log-likelihood of the target documents over all training samples of D:

$$\mathcal{L}_{ce} = \sum_{n}^{N} \sum_{s}^{S_n} log P_\theta(y_{n,s}|y_{n,<s}, x_n) \tag{6}$$

where $y_{n,s}$ is the s-th word of the general audience abstract in the n-th sample, S_n is the length of the target output y_n, and θ gives the parameters of DSPT5. To further improve the performance of DSPT5, we add an additional contrastive loss that forces the prompt encoder to generate the embeddings of extracted keywords from KeyBERT r^{pos} that can be more similar to the representations of subject terms r^{key} provided from the source data, while steering away from negative extracted words r^{neg}, to encourage the model to bring r^{key} and r^{neg} closer in the learned feature space while pushing dissimilar instances r^{key} and r^{neg} apart. Therefore, we propose a new hybrid loss function consisting of the cross-entropy loss in Equ. 6 and a contrastive loss infoNCE (van den Oord et al., 2018) $\mathcal{L}_{nce}$:

$$\mathcal{L}_{nce} = -log \frac{exp(\frac{r^{key} \cdot r^{pos}}{\tau})}{exp(\frac{r^{key} \cdot r^{pos}}{\tau}) + \sum_{r^{neg} \in N^{neg}} exp(\frac{r^{key} \cdot r^{neg}}{\tau})} \tag{7}$$

where r^{key} is from Equ. 2, r^{pos} is from Equ. 3, r^{neg} is from Equ. 4, N^{neg} is the number of negative samples, and τ is a temperature hyperparameter. Details in Appendix Sec. G. The final hybrid loss function for DSPT5 model combines $\mathcal{L}_{ce}$ and $\mathcal{L}_{nce}$:

$$\mathcal{L}_{hybrid} = (1 - \lambda)\mathcal{L}_{ce} + \lambda \mathcal{L}_{nce} \tag{8}$$

where λ is a hyperparameter for $\mathcal{L}_{nce}$.

4.3 Crowd Sampling Decoding

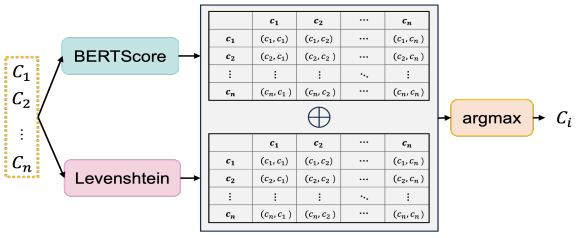

Figure 2: The decoding strategy of crowd sampling.

The decoding objective during inference is to select the best candidate $\hat{y}$ among all possible output

candidates. We first implement a stochastic sampling method (temperature sampling (Ackley et al., 1985)), which selects the next token by sampling from the truncated distribution to generate multiple output candidates. Expanding on Equ. 5:

$$\tilde{P}_\theta(\cdot|x) = \frac{exp(P_\theta(\cdot|x)/\tau)}{\sum exp(P_\theta(\cdot|x)/\tau)} \quad (9)$$

where $\tau \in (0, 1]$ is the temperature parameter. Crowd sampling shows significant performance improvement over standard sampling methods across a wide range of open-ended text generation tasks (Suzgun et al., 2023). To select the final generated output $\hat{y}$, we implement crowd sampling as shown in Figure 2 to get the candidate that can maximize the sum of alignments with the whole crowd by comparing each candidate to the other candidates using the alignment functions. Given a collection of candidate documents $\mathcal{C}$, we get:

$$\hat{y} = \underset{c_i \in \mathcal{C}}{argmax}\Big(\sum_{y_i \in \mathcal{C}} (BERTScore(c_i, y_i)) + \gamma \cdot LEV(c_i, y_i)\Big) \quad (10)$$

where γ is a hyperparameter to adjust the weight for different alignment functions. To measure both the semantic and structural equivalence of the two texts, we consider BERTScore (Zhang* et al., 2020) and Levenshtein similarity (Levenshtein, 1965) as the alignment functions. BERTScore computes the cosine similarity between candidates, considering their representations as vectors. Levenshtein similarity measures the differences between sequences of tokens in the candidates.

5 Experiments

5.1 Experimental Setup

We evaluate the proposed model DSPT5 over eight different colleges in VTechAGP. The dataset is randomly divided into training and testing sets with a ratio of 0.8:0.2 for all other colleges. We discuss the detailed hyperparameters and configurations of DSPT5 in Appendix Sec. C.

5.2 Baselines and Evaluation Metrics

We compare DSPT5 with the following LLM baselines: Pre-trained LLaMA2 (Touvron et al., 2023) [5], ChatGPT (Brown et al., 2020) [6], and Claude2 (Anthropic, 2023); Fine-tuned BART (Lewis et al., 2020), T5 (Raffel et al., 2020), and FLAN-T5 (Chung et al., 2022). Each baseline is fine-tuned on each domain. Crowd sampling is only added to DSPT5 for evaluation, as it is one component of DSPT5. All other LLM baselines use their default decoding strategies. We evaluate DSPT5 with other SOTA LLMs using the following automatic evaluation metrics. (1) Embedding-based metrics: BERTScore (F1) (Zhang* et al., 2020), document-level BLONDE (F1) (Jiang et al., 2022), sentence-level and document-level BLEU (Lin and Och, 2004); (2) Word-based metrics: ROUGE1, ROUGE2 (Lin, 2004) and METEOR (Banerjee and Lavie, 2005); (3) End-to-end metrics: COMET (Rei et al., 2022); (4) Simplicity: SARI (Xu et al., 2016a).(5) Diversity: LCTR (Lexical Translation Consistency Ratio) (Lyu et al., 2021).(6) Readability: FRES (Flesch Reading-Ease Score) (Flesch, 1979); (7) Toxicity (Team et al., 2022) for safety evaluation.

5.3 Results and Analysis

5.3.1 Main Results

The experiment results on VTechAGP are shown in Table 2. We observe that: First, for the accuracy measurement between the generated text and the ground truth (i.e., embedding-based metrics and word-based metrics), our proposed model achieves the best performance in most cases. In addition, fine-tuned models (i.e. Bart, T5, FLAN-T5) always show better translation performance (BLEU, BERTScore, and ROUGE) than pre-trained LLMs (i.e. Claude2, ChatGPTP, LLaMA2). We believe this is because when the reference text (ground truth) is used to update the model's parameters during the fine-tuning process, the model adapts to the nuances and characteristics of our specific academic to general audience text paraphrasing task. In contrast, the pre-trained LLMs are trained on large and diverse datasets and are not optimized for the academic domain. This indicates that it is possible to fine-tune a lightweight language model that can achieve competitive translation performance with LLMs on the high-quality VTechAGP dataset.

Second, pre-trained LLMs outperform fine-tuned lightweight models in end-to-end metrics, simplicity, diversity, and readability. However, different LLMs show distinctive strengths and capabilities. For example, LLaMA2 performs well at text simplification. Claude2 is only good at generating diverse text, and ChatGPT demonstrates the

[5] We implement the 7B version of LLaMA2.
[6] https://chatgpt.com/

Table 2: Performance evaluation of all baselines with twelve different metrics. The abbreviations stand for Agriculture and Life Science, Architecture, Arts and Design, Engineering, Liberal Arts and Human Sciences, Natural Resources and Environment, Science, Business, and Veterinary Medicine.

Model	ALS	AAD	ENG	LAHS	NRE	SCI	BUS	VM	ALS	AAD	ENG	LAHS	NRE	SCI	BUS	VM
	s-BLEU (%)								d-BLEU (%)							
BART	9.25	9.59	10.67	18.01	**12.66**	7.88	14.34	9.51	24.67	32.68	23.88	33.18	26.20	19.51	27.04	20.49
T5	7.93	11.48	8.39	16.63	11.91	7.33	15.35	8.33	**25.61**	32.79	24.82	33.58	23.80	17.00	23.77	19.56
FLAN-T5	8.89	12.27	8.46	17.09	6.94	7.84	14.94	8.33	18.39	28.35	17.45	31.31	17.69	15.88	30.65	18.66
Claude2	1.53	2.08	0.29	2.32	0.83	0.50	2.79	1.19	2.10	8.91	1.06	6.30	2..87	1.33	9.52	3.02
ChatGPT	4.53	5.74	4.34	6.25	4.08	4.20	8.17	4.51	15.39	15.59	13.75	16.99	14.11	12.84	17.20	13.62
LLaMA2	3.39	3.22	3.56	2.39	2.77	2.93	4.02	2.39	16.70	18.51	15.02	18.64	15.51	13.21	15.66	15.54
Ours	**10.84**	**14.09**	**11.25**	**22.12**	9.19	**8.13**	**20.30**	**11.38**	24.95	**33.53**	**24.98**	**38.80**	**28.11**	**21.31**	**35.31**	**23.42**
	BERTScore (F1%)								BLONDE (F1 %)							
BART	85.12	82.83	84.05	86.09	**85.58**	82.74	86.59	83.94	15.04	**45.15**	9.07	44.85	15.22	8.02	38.91	13.64
T5	85.26	83.18	84.26	85.65	84.48	81.70	85.54	83.77	14.64	38.72	8.49	43.14	33.10	7.49	34.47	25.39
FLAN-T5	83.02	81.84	82.26	84.93	82.94	81.22	86.70	82.86	13.42	37.21	7.48	43.02	27.31	7.04	**39.79**	27.98
Claude2	79.48	80.02	77.05	80.95	79.96	77.43	83.25	79.01	10.41	21.13	2.75	17.39	10.37	3.06	21.62	4.94
ChatGPT	84.67	82.48	83.52	84.75	84.65	**83.05**	85.17	83.89	27.73	30.99	**11.80**	30.67	26.70	**11.13**	11.54	21.79
LLaMA2	83.58	80.97	82.46	83.66	83.31	81.46	83.14	82.74	11.38	37.39	8.25	32.77	30.56	7.63	30.01	10.82
Ours	**85.48**	**83.70**	**84.41**	**87.27**	85.51	82.90	**87.97**	**84.20**	**36.75**	44.11	9.62	**48.51**	**36.60**	8.22	35.53	**30.69**
	ROUGE1 (%)								ROUGE2 (%)							
BART	**55.25**	49.86	51.26	57.92	**57.55**	**47.49**	57.78	**50.33**	27.83	27.06	25.98	36.55	**31.16**	20.42	33.75	22.38
T5	53.94	49.74	50.88	55.34	52.07	43.40	52.22	46.66	**29.72**	27.34	25.48	35.08	27.19	18.45	28.64	21.39
FLAN-T5	47.44	45.27	45.22	54.07	47.37	42.48	57.71	46.77	21.54	22.17	19.20	32.30	18.93	16.87	35.75	19.63
Claude2	28.47	36.13	21.95	36.44	33.23	23.13	44.04	29.75	5.42	9.43	2.53	9.47	6.34	3.14	14.63	5.03
ChatGPT	51.50	45.61	47.97	50.32	51.38	46.06	50.35	48.31	20.08	16.75	17.88	21.89	19.67	16.55	21.76	17.54
LLaMA2	44.43	38.30	40.24	39.92	43.22	38.20	41.52	42.49	24.82	23.30	21.86	27.59	24.92	19.37	23.01	22.47
Ours	54.48	**50.84**	**51.73**	**59.67**	56.47	47.14	**60.75**	50.28	28.58	**27.55**	**27.02**	**40.24**	30.83	**21.97**	**39.61**	**24.63**
	METEOR (%)								COMET (%)							
BART	41.27	39.57	39.28	48.65	**45.71**	35.25	44.66	38.28	81.52	75.52	80.76	80.28	82.14	78.56	83.14	80.35
T5	40.34	40.84	39.84	46.86	40.09	31.77	40.04	33.81	80.45	74.77	80.37	76.19	75.66	70.69	77.69	73.18
FLAN-T5	32.97	35.62	33.52	46.04	34.64	31.36	47.86	33.85	77.14	73.26	78.03	78.50	80.02	75.65	82.52	79.04
Claude2	17.50	28.53	14.96	25.81	20.88	15.34	30.85	20.59	74.31	74.38	67.74	75.53	78.02	68.72	82.70	75.05
ChatGPT	38.94	36.86	37.47	41.68	39.71	36.06	40.56	36.09	**85.49**	**80.30**	**84.00**	**83.29**	**85.79**	**82.98**	**85.29**	**84.10**
LLaMA2	**42.53**	39.88	**41.13**	43.68	44.10	**39.51**	41.25	**41.47**	82.11	77.85	82.22	81.17	82.65	79.90	82.88	80.84
Ours	40.50	**42.51**	40.74	**52.67**	45.05	36.65	**50.56**	38.01	81.37	75.90	80.64	81.76	82.09	77.15	84.07	78.36
	SARI								LTCR (%)							
BART	37.22	34.08	36.87	36.34	37.36	36.91	42.43	36.45	59.36	60.85	58.29	55.53	59.97	63.83	61.98	64.02
T5	37.89	35.89	37.88	36.83	36.11	37.21	47.75	36.92	57.61	60.82	56.18	52.27	60.51	62.74	58.84	66.01
FLAN-T5	36.73	34.12	36.78	35.43	34.83	37.76	48.62	36.71	61.41	63.91	60.63	56.89	64.59	**84.12**	61.19	67.16
Claude2	29.34	32.44	28.85	28.48	30.17	31.19	34.79	32.08	**71.66**	**72.94**	**68.61**	**68.46**	**72.21**	73.19	**70.99**	**77.28**
ChatGPT	37.83	35.41	36.90	34.50	36.61	38.53	40.07	38.07	58.87	65.38	57.61	59.64	61.14	61.97	64.28	63.35
LLaMA2	**46.43**	**40.51**	**43.76**	**47.45**	**46.31**	**43.53**	44.99	**44.55**	57.90	61.00	57.90	55.70	61.10	62.56	64.37	62.78
Ours	37.31	36.01	38.21	38.95	37.23	37.11	**48.67**	36.50	56.81	61.33	55.25	51.25	58.35	62.18	56.46	63.46
	FRES								Mean Toxicity (%)							
BART	32.22	40.18	30.60	27.66	39.47	30.80	19.91	29.79	0.06	0.83	0.07	0.71	0.04	0.08	0.09	0.06
T5	32.83	41.60	31.01	27.96	40.69	31.62	20.72	31.82	0.12	0.74	0.08	1.00	0.04	0.11	0.10	0.06
FLAN-T5	40.89	40.89	31.41	29.48	40.99	32.63	22.34	31.62	0.07	0.54	0.08	1.40	0.04	0.11	0.08	0.05
Claude2	31.99	41.97	13.65	21.60	40.45	12.63	26.40	29.45	0.06	0.49	0.10	0.93	0.09	0.11	0.10	0.23
ChatGPT	**42.11**	**42.41**	**33.85**	**31.72**	**41.70**	**34.15**	**29.69**	**42.21**	**0.03**	**0.06**	0.05	**0.32**	0.03	**0.05**	0.09	**0.03**
LLaMA2	34.26	41.60	31.92	30.70	33.54	32.73	22.85	32.53	0.11	0.55	0.16	1.06	0.08	0.13	0.13	0.11
Ours	33.65	41.50	31.62	28.17	41.19	32.33	21.74	33.34	0.12	1.19	**0.04**	0.97	**0.02**	0.06	**0.04**	**0.03**

Table 3: Ablation study over DSPT5 components on VTechAGP across eight colleges. $\text{DSPT5}_{\text{dynamic}}$ is the model only using a dynamic hard prompt from keyBERT. $\text{DSPT5}_{\text{dyn+soft}}$ is the model using a dynamic soft prompt from the prompt encoder. $\text{DSPT5}_{\text{dyn+soft+con}}$ is the model fine-tuned using dynamic soft prompt with contrastive learning loss. $\text{DSPT5}_{\text{all}}$ is the final version of the model using dynamic soft prompt with contrastive learning for training and crowd sampling for inference.

Model	s-BLEU	d-BLEU	BERTScore	BLONDE	ROUGE1	ROUGE2	METEOR	COMET	SARI	LTCR	FRES	Toxicity
$\text{T5}_{\text{pre-trained}}$	3.76	0.84	78.61	11.34	23.74	12.06	11.74	59.66	29.81	**78.01**	**36.36**	0.23
$\text{T5}_{\text{fine-tuned}}$	6.02	19.14	82.14	22.14	45.53	22.14	36.44	68.74	36.55	62.77	31.54	0.20
$\text{Ours}_{\text{dynamic}}$	8.78	22.83	83.83	28.43	49.77	25.33	38.35	77.28	36.71	60.19	32.31	0.24
$\text{Ours}_{\text{dyn+soft}}$	9.87	25.79	84.43	34.92	51.14	26.98	40.64	78.91	37.03	59.11	32.34	**0.11**
$\text{Ours}_{\text{dyn+soft+con}}$	9.88	26.78	**84.58**	33.99	**51.66**	26.82	40.90	78.54	37.03	59.05	32.33	0.13
Ours_{all}	**9.93**	**26.99**	84.52	**35.75**	51.45	**27.59**	**41.40**	**79.06**	**37.05**	58.77	32.39	0.17

best performance for end-to-end metrics and readability. Each LLM has a unique architecture and has been pre-trained on different datasets, which may explain the observed differences. Finding an LLM that performs well in all evaluation metrics across all domains is challenging, indicating that there is still much room for improvement in LLMs for academic-to-general-audience text paraphrasing tasks. Besides, we need to further explore new evaluation metrics that can more accurately reflect the quality of the generated general audience text.

5.3.2 Ablation Study

To assess the performance of each component in DSPT5, we conducted the ablation study. For each run, we randomly sampled 15 data points from each college to construct a balanced dataset for the ablation study (see ablation study setup in Appendix C). The results are presented in Table 3 as the mean value of five different subsets. We observe that: (1) Fine-tuning the LLM can significantly improve its performance. This is because fine-tuning the model on a customized dataset can enable it to learn domain-specific nuances and identify task-specific patterns. (2) Adding the dynamic soft prompt consisting of domain-specific keywords can further improve the performance. The results verify that adding the extracted academic keywords with the phrase "reduce the domain knowledge of" in the prompt template while training with contrastive-generative loss can reduce the domain knowledge. Besides, the crowd sampling decoding is only implemented in the inference stage, it is worth adding the module for further performance improvement; (3) The diversity LTCR and readability FRES decrease after fine-tuning T5. We suspect that this is because T5 is pre-trained on diverse and large datasets, resulting in the generated text being represented by a diverse vocabulary. Moreover, the output of $T5_{\text{pre-trained}}$ is always shortened compared to $T5_{\text{fine-tuned}}$, making the output easier to understand (higher FRES). Thus, new evaluation metrics for academic-to-general-audience text paraphrasing tasks are needed in the future.

5.3.3 Human Evaluation

Following a similar setting as (Liu et al., 2024; Li et al., 2024; Song et al., 2024; Kew et al., 2023; Devaraj et al., 2021), our evaluation uses a random sample of 20 abstracts from the test split VTechAGP considering the workload. Judges are presented with both the academic abstract and generated general-audience abstracts from seven models for each data sample in a total of 140 abstracts. Using a 1-5 Likert scale, the judges are asked to rate the model output based on five criteria: comprehensiveness, layness, meaning preservation, conciseness, and fluency. More details are discussed in detail in Sec. F in the Appendix.

Table 4: Mean human evaluation ratings 1-5 (the higher the better) of different models on VTechAGP. Reported from left to right are: comprehensiveness, layness, meaning preservation, conciseness, and fluency. ICC stands for the intraclass correlation coefficient.

	COM	LAY	MP	CON	FLU	ICC
BART	3.83	2.83	3.67	3.58	3.83	**0.85**
T5	3.61	2.95	3.35	3.5	3.51	0.60
FlanT5	3.71	3.13	3.41	3.66	3.76	0.38
Claude2	3.08	2.95	2.72	3.55	3.26	0.67
ChatGPT	**4.17**	**3.66**	3.83	3.71	**4.31**	**0.85**
LLaMA2	4.05	3.61	3.73	3.80	**4.31**	0.39
Ours	**4.17**	3.57	**3.90**	**3.87**	3.95	0.78

Human evaluation results are shown in Table 4. From Table 4, we observe that generally ChatGPT, and our proposed DSPT5 show better performance than other baselines. ChatGPT performs well in comprehensiveness, layness, and fluency, whereas DSPT5 outperforms all other baselines in comprehensiveness, meaning perservation, and conciseness. Based on the observations, new information with simpler words is always introduced in the generated outputs from ChatGPT. Those sentences with simpler words make the generated abstracts easier to understand, resulting in better layness. However, it also significantly changes the meaning of the original sentences and makes the sentences longer with redundancy. That's why ChatGPT shows the best performance in layness, while not the best in meaning preservation and conciseness. DSPT5 attempts to paraphrase the abstracts with non-technical words while retaining the original meaning as closely as possible. While keeping the content concise, it may sacrifice fluency between sentences.

For intraclass correlation coefficient (ICC) [7], which is used to determine if items or subjects can be rated reliably by different raters, BART, ChatGPT and DSPT5 show good reliability. T5 and Claude2 show moderate reliability. FlanT5 and LLaMA2 show poor reliability according to (Koo and Li, 2016). Although ChatGPT shows com-

[7]We use Pingouin package to calculate ICC: https://pingouin-stats.org/build/html/index.html

petitive results, it is NOT an open-sourced model, which may raise concerns about data security and high-cost issues. Alternatively, the DSPT5 can be fully controlled and owned with only a single V100 GPU for fine-tuning jobs for the academic-to-general-audience text paraphrasing task.

6 Conclusion

We created VTechAGP [8], an academic-to-general-audience document-level translation benchmark dataset, which consists of academic and general audience abstract pairs with their corresponding auxiliary information such as title, subject terms, etc. We explore several SOTA LLMs to establish the baseline performance on the text paraphrasing task. We propose a new model, DSPT5, which includes dynamic soft prompt generation with hybrid loss during the training phase and a new crowd-sampling strategy in the inference stage. We evaluate the datasets and models from the perspective of document-level embedding-based, word-based, end-to-end metrics, simplicity, diversity, readability, toxicity, and human evaluation. Extensive experimental results on 8 different colleges of VTechAGP show that DSPT5 achieves comparable results with other SOTA benchmark models.

7 Limitations

We identify the following limitations (remaining challenges) and future directions:

(1) Dataset: Although VTechAGP is a recent and distinctive dataset, it suffers in terms of size. This is due to the fact that the policy of including a general audience version of the abstract for all ETDs submitted to VTechWorks began to be implemented in 2016. However, this aspect of the limitation should be gradually overcome as VTechWorks continues to collect and maintain this collection of documents. With the increase in graduate student enrollment at Virginia Tech in recent years, it is safe to assume that VTechAGP will continue to grow in the coming years. In addition, VTechAGP consists of pairs of abstracts carefully written by students, encapsulating their work during their time in graduate school. With about 50% of the corpus composed of Ph.D. dissertations and the other half composed of M.S. theses, we believe that VTechAGP should achieve good quality due to the amount of time and effort put into the creation process. As the size of the dataset grows, VTechAGP will become more robust for text paraphrasing tasks in the near future.

(2) Model: While transformer-based LLMs retain their popularity for text generation tasks, our experiment results in Sec. 5 show that there is still a huge room for further improvement of LLMs in academic to general audience text paraphrasing tasks. With the rapid development of LLMs, more robust and efficient LLMs are emerging. For example, the recently released model Gemini (Team et al., 2023) shows great potential and competitive results compared to ChatGPT and LLaMA2 in natural language understanding-related tasks. In addition, some recently released lightweight decoder-only models (e.g., Phi-2 (Li et al., 2023)) also exhibit great potential for text generation tasks. As the VTechAGP dataset grows, it will be able to support fine-tuning for more powerful open-source LLMs.

(3) Evaluation Metric: Although we have implemented twelve different automatic evaluation metrics, including document-level embedding-based metrics, word-based metrics, end-to-end metrics, simplicity, diversity, readability, and toxicity, we still lack a representative metric that can evaluate the quality of the generated text in terms of general audience understanding while remaining scientifically accurate and easy to comprehend. The existing evaluation metrics for simplicity such as SARI (Xu et al., 2016a) focus on measuring the goodness of words that are added, deleted, and kept by the system. In addition, regarding the formula of FRES (Flesch, 1979), such readability evaluation metric only considers the number of words and the number of sentences when evaluating the ease of understanding for children of different grades. Instead of just simplifying sentence structure and using words with fewer letters, our task aims to reduce the domain knowledge of the generated text while remaining scientifically correct, so that the general audience or people/researchers in other research fields (departments) can still understand the core idea for further interdisciplinary collaboration. New automatic evaluation metrics or large-scale human evaluation approaches for academic-to-general-audience text paraphrasing tasks need to be explored in future work.

Acknowledgments

This project was made possible in part by the Institute of Museum and Library Services (LG-256638-OLS-24).

[8] Benchmarks: https://github.com/SIGSEGV-0x7/VTechAGP-Benchmark

References

David H Ackley, Geoffrey E Hinton, and Terrence J Sejnowski. 1985. A learning algorithm for Boltzmann machines. *Cognitive science*, 9(1):147–169.

Fernando Alva-Manchego, Louis Martin, Antoine Bordes, Carolina Scarton, Benoît Sagot, and Lucia Specia. 2020. ASSET: A dataset for tuning and evaluation of sentence simplification models with multiple rewriting transformations. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 4668–4679, Online. Association for Computational Linguistics.

Chenxin An, Jiangtao Feng, Kai Lv, Lingpeng Kong, Xipeng Qiu, and Xuanjing Huang. 2022. Cont: Contrastive neural text generation. *Advances in Neural Information Processing Systems*, 35:2197–2210.

Anthropic. 2023. Model card and evaluations for claude models.

Dennis Aumiller and Michael Gertz. 2022. Klexikon: A German dataset for joint summarization and simplification. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 2693–2701, Marseille, France. European Language Resources Association.

Satanjeev Banerjee and Alon Lavie. 2005. METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In *Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization*, pages 65–72, Ann Arbor, Michigan. Association for Computational Linguistics.

Sofia Blinova, Xinyu Zhou, Martin Jaggi, Carsten Eickhoff, and Seyed Ali Bahrainian. 2023. SIMSUM: Document-level text simplification via simultaneous summarization. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 9927–9944.

Ondrej Bojar, Rajen Chatterjee, Christian Federmann, Yvette Graham, Barry Haddow, Matthias Huck, Antonio Jimeno Yepes, Philipp Koehn, Varvara Logacheva, Christof Monz, et al. 2016. Findings of the 2016 conference on machine translation (wmt16). In *First Conference on Machine Translation*, pages 131–198. Association for Computational Linguistics.

Samuel R Bowman, Gabor Angeli, Christopher Potts, and Christopher D Manning. 2015. A large annotated corpus for learning natural language inference. *arXiv preprint arXiv:1508.05326*.

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. *Advances in neural information processing systems*, 33:1877–1901.

Daniel Cer, Mona Diab, Eneko Agirre, Iñigo Lopez-Gazpio, and Lucia Specia. 2017. SemEval-2017 task 1: Semantic textual similarity multilingual and crosslingual focused evaluation. In *Proceedings of the 11th International Workshop on Semantic Evaluation (SemEval-2017)*, pages 1–14, Vancouver, Canada. Association for Computational Linguistics.

Derek Chen, Celine Lee, Yunan Lu, Domenic Rosati, and Zhou Yu. 2023. Mixture of soft prompts for controllable data generation. *arXiv preprint arXiv:2303.01580*.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, et al. 2022. Scaling instruction-finetuned language models. *arXiv preprint arXiv:2210.11416*.

Raj Dabre, Chenhui Chu, and Anoop Kunchukuttan. 2020. A survey of multilingual neural machine translation. *ACM Computing Surveys (CSUR)*, 53(5):1–38.

Sumanth Dathathri, Andrea Madotto, Janice Lan, Jane Hung, Eric Frank, Piero Molino, Jason Yosinski, and Rosanne Liu. 2020. Plug and play language models: A simple approach to controlled text generation. In *International Conference on Learning Representations*.

Ashwin Devaraj, Iain Marshall, Byron Wallace, and Junyi Jessy Li. 2021. Paragraph-level simplification of medical texts. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4972–4984, Online. Association for Computational Linguistics.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

William B. Dolan and Chris Brockett. 2005. Automatically constructing a corpus of sentential paraphrases. In *Proceedings of the Third International Workshop on Paraphrasing (IWP2005)*.

Anthony Fader, Luke Zettlemoyer, and Oren Etzioni. 2013. Paraphrase-driven learning for open question answering. In *Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1608–1618.

Akhbardeh Farhad, Arkhangorodsky Arkady, Biesialska Magdalena, Bojar Ondřej, Chatterjee Rajen, Chaudhary Vishrav, Marta R Costa-jussa, España-Bonet Cristina, Fan Angela, Federmann Christian, et al. 2021. Findings of the 2021 conference on machine translation (WMT21). In *Proceedings of the Sixth*

Conference on Machine Translation, pages 1–88. Association for Computational Linguistics.

Rudolf Flesch. 1979. How to write plain English. *University of Canterbury. Available at http://www. mang. canterbury. ac. nz/writing_guide/writing/flesch. shtml.[Retrieved 5 February 2016]*.

Lorenzo Jaime Flores, Heyuan Huang, Kejian Shi, Sophie Chheang, and Arman Cohan. 2023. Medical text simplification: Optimizing for readability with unlikelihood training and reranked beam search decoding. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 4859–4873, Singapore. Association for Computational Linguistics.

Juri Ganitkevitch, Benjamin Van Durme, and Chris Callison-Burch. 2013. PPDB: The paraphrase database. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 758–764, Atlanta, Georgia. Association for Computational Linguistics.

Maarten Grootendorst. 2020. KeyBERT: Minimal keyword extraction with BERT.

Jiaxian Guo, Sidi Lu, Han Cai, Weinan Zhang, Yong Yu, and Jun Wang. 2018. Long text generation via adversarial training with leaked information. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32.

Freya Hewett and Manfred Stede. 6. Automatically evaluating the conceptual complexity of German texts. In *Proceedings of the 17th Conference on Natural Language Processing (KONVENS 2021)*, pages 228–234, Düsseldorf, Germany. KONVENS 2021 Organizers.

Ari Holtzman, Jan Buys, Li Du, Maxwell Forbes, and Yejin Choi. 2020. The curious case of neural text degeneration. In *International Conference on Learning Representations*.

Kuan-Hao Huang, Varun Iyer, I-Hung Hsu, Anoop Kumar, Kai-Wei Chang, and Aram Galstyan. 2023. ParaAMR: A large-scale syntactically diverse paraphrase dataset by AMR back-translation. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 8047–8061, Toronto, Canada. Association for Computational Linguistics.

Chao Jiang, Mounica Maddela, Wuwei Lan, Yang Zhong, and Wei Xu. 2020. Neural CRF model for sentence alignment in text simplification. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 7943–7960. Association for Computational Linguistics.

Gongyao Jiang, Xinran Shi, and Qiong Luo. 2024. Llm-collaboration on automatic science journalism for the general audience. *arXiv preprint arXiv:2407.09756*.

Yuchen Jiang, Tianyu Liu, Shuming Ma, Dongdong Zhang, Jian Yang, Haoyang Huang, Rico Sennrich, Ryan Cotterell, Mrinmaya Sachan, and Ming Zhou. 2022. BlonDe: An automatic evaluation metric for document-level machine translation. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 1550–1565, Seattle, United States. Association for Computational Linguistics.

Tannon Kew, Alison Chi, Laura Vásquez-Rodríguez, Sweta Agrawal, Dennis Aumiller, Fernando Alva-Manchego, and Matthew Shardlow. 2023. BLESS: Benchmarking large language models on sentence simplification. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 13291–13309, Singapore. Association for Computational Linguistics.

Philipp Koehn, Franz Josef Och, and Daniel Marcu. 2003. Statistical phrase-based translation. In *Proceedings of the 2003 Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics*, pages 127–133.

Terry K Koo and Mae Y Li. 2016. A guideline of selecting and reporting intraclass correlation coefficients for reliability research. *Journal of chiropractic medicine*, 15(2):155–163.

Carl Lagoze, Herbert Van de Sompel, Michael Nelson, and Simeon Warner. 2002. The Open Archives Initiative - Protocol for Metadata Harvesting - v.2.0. Protocol Version 2.0.

Wuwei Lan, Siyu Qiu, Hua He, and Wei Xu. 2017. A continuously growing dataset of sentential paraphrases. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 1224–1234, Copenhagen, Denmark. Association for Computational Linguistics.

Chia-Hsuan Lee, Aditya Siddhant, Viresh Ratnakar, and Melvin Johnson. 2022. DOCmT5: Document-level pretraining of multilingual language models. In *Findings of the Association for Computational Linguistics: NAACL 2022*, pages 425–437, Seattle, United States. Association for Computational Linguistics.

Vladimir I. Levenshtein. 1965. Binary codes capable of correcting deletions, insertions, and reversals. *Soviet physics. Doklady*, 10:707–710.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online. Association for Computational Linguistics.

Hao Li, Yuping Wu, Viktor Schlegel, Riza Batista-Navarro, Tharindu Madusanka, Iqra Zahid, Jiayan Zeng, Xiaochi Wang, Xinran He, Yizhi Li, and Goran Nenadic. 2024. Which side are you on? a multi-task dataset for end-to-end argument summarisation and evaluation. In *Findings of the Association for Computational Linguistics ACL 2024*, pages 133–150, Bangkok, Thailand and virtual meeting. Association for Computational Linguistics.

Xiang Li, John Thickstun, Ishaan Gulrajani, Percy S Liang, and Tatsunori B Hashimoto. 2022. Diffusion-lm improves controllable text generation. *Advances in Neural Information Processing Systems*, 35:4328–4343.

Yuanzhi Li, Sébastien Bubeck, Ronen Eldan, Allie Del Giorno, Suriya Gunasekar, and Yin Tat Lee. 2023. Textbooks are all you need ii: Phi-1.5 technical report. *arXiv preprint arXiv:2309.05463*.

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

Chin-Yew Lin and Franz Josef Och. 2004. ORANGE: A method for evaluating automatic evaluation metrics for machine translation. In *COLING 2004: Proceedings of the 20th International Conference on Computational Linguistics*, pages 501–507, Geneva, Switzerland. COLING.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. 2014. Microsoft coco: Common objects in context. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part V 13*, pages 740–755. Springer.

Ran Liu, Ming Liu, Min Yu, He Zhang, Jianguo Jiang, Gang Li, and Weiqing Huang. 2024. SumSurvey: An abstractive dataset of scientific survey papers for long document summarization. In *Findings of the Association for Computational Linguistics ACL 2024*, pages 9632–9651, Bangkok, Thailand and virtual meeting. Association for Computational Linguistics.

Haotian Luo, Yixin Liu, Peidong Liu, and Xianggen Liu. 2023. Vector-quantized prompt learning for paraphrase generation. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 13389–13398, Singapore. Association for Computational Linguistics.

Junyu Luo, Junxian Lin, Chi Lin, Cao Xiao, Xinning Gui, and Fenglong Ma. 2022a. Benchmarking automated clinical language simplification: Dataset, algorithm, and evaluation. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 3550–3562, Gyeongju, Republic of Korea. International Committee on Computational Linguistics.

Junyu Luo, Junxian Lin, Chi Lin, Cao Xiao, Xinning Gui, and Fenglong Ma. 2022b. Benchmarking automated clinical language simplification: Dataset, algorithm, and evaluation. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 3550–3562.

Xinglin Lyu, Junhui Li, Zhengxian Gong, and Min Zhang. 2021. Encouraging lexical translation consistency for document-level neural machine translation. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 3265–3277, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Tania Josephine Martin, José Ignacio Abreu Salas, and Paloma Moreda Pozo. 2023. A review of parallel corpora for automatic text simplification. Key challenges moving forward. In *International Conference on Applications of Natural Language to Information Systems*, pages 62–78. Springer.

Philip M McCarthy. 2005. *An Assessment of the Range and Usefulness of Lexical Diversity Measures and the Potential of the Measure of Textual, Lexical Diversity (MTLD)*. Ph.D. thesis, The University of Memphis.

Babak Naderi, Salar Mohtaj, Kaspar Ensikat, and Sebastian Möller. 2019. Subjective assessment of text complexity: A dataset for german language. *arXiv preprint arXiv:1904.07733*.

Jonathan Pei, Kevin Yang, and Dan Klein. 2023. PREADD: Prefix-adaptive decoding for controlled text generation. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 10018–10037, Toronto, Canada. Association for Computational Linguistics.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *The Journal of Machine Learning Research*, 21(1):5485–5551.

Surangika Ranathunga, En-Shiun Annie Lee, Marjana Prifti Skenduli, Ravi Shekhar, Mehreen Alam, and Rishemjit Kaur. 2023. Neural machine translation for low-resource languages: A survey. *ACM Computing Surveys*, 55(11):1–37.

Ricardo Rei, Ana C Farinha, José G.C. de Souza, Pedro G. Ramos, André F.T. Martins, Luisa Coheur, and Alon Lavie. 2022. Searching for COMETINHO: The little metric that could. In *Proceedings of the 23rd Annual Conference of the European Association for Machine Translation*, pages 61–70, Ghent, Belgium. European Association for Machine Translation.

Scarton Scarton, Mikel L. Forcada, Miquel Esplà-Gomis, and Lucia Specia. 11 2-3 2019. Estimating post-editing effort: A study on human judgements, task-based and reference-based metrics of MT quality. In *Proceedings of the 16th International Conference on Spoken Language Translation*, Hong Kong. Association for Computational Linguistics.

Arwinder Singh and Gurpreet Singh Josan. 2022. Paraphrase generation: A review from RNN to transformer based approaches. *International Journal of Next-Generation Computing*.

Hwanjun Song, Hang Su, Igor Shalyminov, Jason Cai, and Saab Mansour. 2024. FineSurE: Fine-grained summarization evaluation using LLMs. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 906–922, Bangkok, Thailand. Association for Computational Linguistics.

Regina Stodden, Omar Momen, and Laura Kallmeyer. 2023. DEplain: A German parallel corpus with intralingual translations into plain language for sentence and document simplification. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 16441–16463, Toronto, Canada. Association for Computational Linguistics.

Renliang Sun, Wei Xu, and Xiaojun Wan. 2023. Teaching the pre-trained model to generate simple texts for text simplification. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 9345–9355, Toronto, Canada. Association for Computational Linguistics.

Mirac Suzgun, Luke Melas-Kyriazi, and Dan Jurafsky. 2023. Follow the wisdom of the crowd: Effective text generation via minimum Bayes risk decoding. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 4265–4293, Toronto, Canada. Association for Computational Linguistics.

Gemini Team, Rohan Anil, Sebastian Borgeaud, Yonghui Wu, Jean-Baptiste Alayrac, Jiahui Yu, Radu Soricut, Johan Schalkwyk, Andrew M Dai, Anja Hauth, et al. 2023. Gemini: A family of highly capable multimodal models. *arXiv preprint arXiv:2312.11805*.

NLLB Team, Marta R. Costa-jussà, James Cross, Onur Çelebi, Maha Elbayad, Kenneth Heafield, Kevin Heffernan, Elahe Kalbassi, Janice Lam, Daniel Licht, Jean Maillard, Anna Sun, Skyler Wang, Guillaume Wenzek, Al Youngblood, Bapi Akula, Loic Barrault, Gabriel Mejia Gonzalez, Prangthip Hansanti, John Hoffman, Semarley Jarrett, Kaushik Ram Sadagopan, Dirk Rowe, Shannon Spruit, Chau Tran, Pierre Andrews, Necip Fazil Ayan, Shruti Bhosale, Sergey Edunov, Angela Fan, Cynthia Gao, Vedanuj Goswami, Francisco Guzmán, Philipp Koehn, Alexandre Mourachko, Christophe Ropers, Safiyyah Saleem, Holger Schwenk, and Jeff Wang. 2022. No Language Left Behind: Scaling Human-Centered Machine Translation. *Preprint*, arxiv:2207.04672.

Jörg Tiedemann. 2020. The tatoeba translation Challenge–Realistic data sets for low resource and multilingual MT. *arXiv preprint arXiv:2010.06354*.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. 2023. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

Sowmya Vajjala and Ivana Lučić. 2018a. OneStopEnglish corpus: A new corpus for automatic readability assessment and text simplification. In *Proceedings of the Thirteenth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 297–304, New Orleans, Louisiana. Association for Computational Linguistics.

Sowmya Vajjala and Ivana Lučić. 2018b. OneStopEnglish corpus: A new corpus for automatic readability assessment and text simplification. In *Proceedings of the Thirteenth Workshop on Innovative Use of NLP for Building Educational Applications*, pages 297–304.

Aaron van den Oord, Yazhe Li, and Oriol Vinyals. 2018. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

Yixin Wan, Kuan-Hao Huang, and Kai-Wei Chang. 2023. PIP: Parse-instructed prefix for syntactically controlled paraphrase generation. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 10372–10380, Toronto, Canada. Association for Computational Linguistics.

Zhiguo Wang, Wael Hamza, and Radu Florian. 2017. Bilateral multi-perspective matching for natural language sentences. In *Proceedings of the Twenty-Sixth International Joint Conference on Artificial Intelligence, IJCAI-17*, pages 4144–4150.

John Wieting and Kevin Gimpel. 2017. ParaNMT-50M: Pushing the limits of paraphrastic sentence embeddings with millions of machine translations. *arXiv preprint arXiv:1711.05732*.

Jiacheng Xu, Caiming Xiong, Silvio Savarese, and Yingbo Zhou. 2022. Best- k search algorithm for neural text generation. *arXiv preprint arXiv:2211.11924*.

Wei Xu, Chris Callison-Burch, and Courtney Napoles. 2015. Problems in current text simplification research: New data can help. *Transactions of the Association for Computational Linguistics*, 3:283–297.

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

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

Kexin Yang, Dayiheng Liu, Wenqiang Lei, Baosong Yang, Mingfeng Xue, Boxing Chen, and Jun Xie. 2023. Tailor: A soft-prompt-based approach to attribute-based controlled text generation. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 410–427, Toronto, Canada. Association for Computational Linguistics.

Ruslan Yermakov, Nicholas Drago, and Angelo Ziletti. 2021. Biomedical data-to-text generation via fine-tuning transformers. In *Proceedings of the 14th International Conference on Natural Language Generation*, pages 364–370, Aberdeen, Scotland, UK. Association for Computational Linguistics.

Hokeun Yoon and JinYeong Bak. 2023. Diversity enhanced narrative question generation for storybooks. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 465–482, Singapore. Association for Computational Linguistics.

Farooq Zaman, Faisal Kamiran, Matthew Shardlow, Saeed Ul Hassan, Asim Karim, and Naif Radi Aljohani. Sats: Simplification aware text summarisation of scientific documents. *Frontiers in Artificial Intelligence*, 7:1375419.

Biao Zhang, Philip Williams, Ivan Titov, and Rico Sennrich. 2020. Improving massively multilingual neural machine translation and zero-shot translation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1628–1639, Online. Association for Computational Linguistics.

Tianyi Zhang*, Varsha Kishore*, Felix Wu*, Kilian Q. Weinberger, and Yoav Artzi. 2020. BERTScore: Evaluating text generation with BERT. In *International Conference on Learning Representations*.

Xingxing Zhang and Mirella Lapata. 2017. Sentence simplification with deep reinforcement learning. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*, pages 584–594, Copenhagen, Denmark. Association for Computational Linguistics.

Yizhe Zhang, Guoyin Wang, Chunyuan Li, Zhe Gan, Chris Brockett, and Bill Dolan. 2020. POINTER: Constrained progressive text generation via insertion-based generative pre-training. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 8649–8670, Online. Association for Computational Linguistics.

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

Michał Ziemski, Marcin Junczys-Dowmunt, and Bruno Pouliquen. 2016. The united nations parallel corpus v1. 0. In *Proceedings of the Tenth International Conference on Language Resources and Evaluation (LREC'16)*, pages 3530–3534.

A Dataset Comparison

In this section, we explore different features (i.e. document-level or sentence-level, multiple domains or single domain, with auxiliary information or not, in English or other languages) of existing public datasets for text simplification and text paraphrasing, which motivated us to collect and publish a new academic-to-general-audience text paraphrasing dataset VTechAGP, which is a document-level, multi-domain English text paraphrasing dataset with domain-specific auxiliary information. We show the comparison of our proposed dataset VTechAGP with other existing public datasets. The details of features are displayed in Table 5.

B Dataset Analysis

The original dataset does not include the college information. We manually assign each text sample to the college subset based on the department information on each official college website from Virginia Tech. Such data preprocessing steps introduce a few duplicates in the data because: (1) Some departments are involved in more than one college. For example, according to Virginia Tech's official website, the Department of Biology Systems Engineering appears both in the College of Engineering [9] and the College of Agriculture and Life Sciences [10]. (2) According to VT each College's website, some departments (i.e. Environmental Science and Engineering, Counselor Education, etc.) can not be assigned to a specific college. We use fuzzy matches to find similar but not exact matches to assign data to colleges. Note that such a processing step may introduce a small number of duplicate files.

After processing, VTechAGP contains a total of 4,938 document pairs (academic abstract and general audience abstract), where 52.8% of the documents are from Ph.D. dissertations and 47.2% of the documents are from Master's theses. Table 1

[9] https://eng.vt.edu/academics/departments.html

[10] https://www.cals.vt.edu/departments-and-school.html

Table 5: Comparison between VTechAGP and other text simplification or paraphrase datasets. Columns indicate whether the dataset has the feature.

	Doc.	Multi Domains	Side Info.	Eng.
Newsela (Xu et al., 2015)	✓	✗	✗	✓
WikiSmall (Zhu et al., 2010)	✗	✗	✗	✓
WikiLarge (Zhang and Lapata, 2017)	✗	✗	✗	✓
WikiAuto (Jiang et al., 2020)	✓	✗	✓	✓
TurkCorpus (Xu et al., 2016b)	✗	✗	✗	✓
ASSET (Alva-Manchego et al., 2020)	✗	✗	✗	✓
MedLane (Luo et al., 2022b)	✗	✗	✗	✓
OneStopEnglish (Vajjala and Lučić, 2018b)	✗	✗	✗	✓
Lexica (Hewett and Stede, 6)	✓	✗	✓	✗
DEplain (Stodden et al., 2023)	✓	✓	✓	✗
Quora (Wang et al., 2017)	✗	✗	✗	✓
Paralex (Fader et al., 2013)	✗	✗	✗	✓
MRPC (Dolan and Brockett, 2005)	✗	✗	✗	✓
STS (Cer et al., 2017)	✗	✗	✗	✓
SNLI (Bowman et al., 2015)	✗	✗	✓	✓
ParaNMT-50M (Wieting and Gimpel, 2017)	✗	✗	✗	✓
MSCOCO (Lin et al., 2014)	✗	✗	✓	✓
VTechAGP (ours)	✓	✓	✓	✓

Table 6: Description of CSV File Columns in the Dataset

Column Name	Description
identifier_uri	Persistent identifier (CNRI handle)
title	Title
abstract	Regular abstract
abstract_general	General audience abstract
subject_terms	List of subject terms
discipline	Field of study for the degree awarded
department	Name of the academic department
degree	Degree awarded
degree_level	Degree level ('doctoral' or 'masters')
type	Type of ETD ('thesis' or 'dissertation')

shows the data statistics of VTechAGP across eight different colleges. VTechAGP is divided into eight colleges because data from different colleges exhibits distinct statistics. In addition, the dataset divided into colleges can also support potential cross-domain research tasks and provide convenience for researchers focusing on a specific domain in the future. The document pairs are categorized based on colleges since authors from the same college may share some common terminology and background knowledge. Therefore, these categories should exhibit similar statistics. Table 1 reports some basic dataset statistics, including the number of documents for each college, the average number of sentences for each document, and the average sentence length for the source and target texts, respectively. We also report textual and lexical diversity (MTLD), which reflects the average number of words in a row for which a certain type-token ratio is maintained (McCarthy, 2005). To better illustrate the difference in readability between the source and target texts, we report Readability Consensus,[11] which is a combined evaluation of Flesch Kincaid Grade, Flesch Reading Ease, SMOG Index, Coleman-Liau Index, Automated Readability Index, Dale-Chall Readability Score, Linsear Write Formula, and Gunning FOG Formula. From Table 1, we observe that the general audience abstracts have fewer sentences and shorter sentence lengths compared to the academic abstracts. Furthermore, academic abstracts have better lexical diversity than general audience abstracts, except for the College of Architecture, Arts and Design, and the College of Business. In general, academic abstracts have a higher level of comprehension difficulty compared to general audience abstracts, except the College of Engineering. Analysis for the College of Engineering, which represents almost half of the total dataset, is discussed below.

The distinct category (College of Engineering) contains the largest number of document pairs, representing almost half of the total dataset, due to

[11] https://pypi.org/project/textstat/

Table 7: An example of academic abstract and general-audience abstract pair.

Academic Abstract
Breath-first search (BFS) is a fundamental building block in many graph-based applications, but it is difficult to optimize for a field-programmable gate array (FPGA) due to its irregular memory-access patterns. Prior work, based on hardware description languages (HDLs) and high-level synthesis (HLS), addresses the memory-access bottleneck of BFS by using techniques such as data alignment and compute-unit replication on FPGAs. The efficacy of such optimizations depends on factors such as the sparsity of target graph datasets. Optimizations intended for sparse graphs may not work as effectively for dense graphs on an FPGA and vice versa. This thesis presents two sets of FPGA optimization strategies for BFS, one for near-hypersparse graphs and the other designed for sparse to moderately dense graphs. For near-hypersparse graphs, a queue-based kernel with maximal use of local memory on FPGA is implemented. For denser graphs, an array-based kernel with compute-unit replication is implemented. Across a diverse collection of graphs, our OpenCL optimization strategies for near-hypersparse graphs delivers a 5.7x to 22.3x speedup over a state-of-the-art OpenCL implementation, when evaluated on an Intel Stratix 10 FPGA. The optimization strategies for sparse to moderately dense graphs deliver 1.1x to 2.3x speedup over a state-of-the-art OpenCL implementation on the same FPGA. Finally, this work uses graph metrics such as average degree and Gini coefficient to observe the impact of graph properties on the performance of the proposed optimization strategies.
General-Audience Abstract
A graph is a data structure that typically consists of two sets – a set of vertices and a set of edges representing connections between the vertices. Graphs are used in a broad set of application domains such as the testing and verification of digital circuits, data mining of social networks, and analysis of road networks. In such application areas, breadth-first search (BFS) is a fundamental building block. BFS is used to identify the minimum number of edges needed to be traversed from a source vertex to one or many destination vertices. In recent years, several attempts have been made to optimize the performance of BFS on reconfigurable architectures such as field-programmable gate arrays (FPGAs). However, the optimization strategies for BFS are not necessarily applicable to all types of graphs. Moreover, the efficacy of such optimizations oftentimes depends on the sparsity of input graphs. To that end, this work presents optimization strategies for graphs with varying levels of sparsity. Furthermore, this work shows that by tailoring the BFS design based on the sparsity of the input graph, significant performance improvements are obtained over the state-of-the-art BFS implementations on an FPGA.

Table 8: Dataset statistics of VTechAGP over 13 departments in the College of Engineering. Statistics are reported in the format of academic (source) / general audience (target) doc.

Department	Readability Consensus
Aerospace Eng.	14th-15th / 14th-15th
Biomedical Eng.	13th-14th / 10th-11th
Building Construction	15th-16th / 15th-16th
Chemical Eng.	16th-17th / 11th-12th
Civil Eng.	16th-17th / 15th-16th
Computer Sci.	14th-15th / 15th-16th
Electrical & Computer Eng.	15th-16th / 15th-16th
Eng. Education	17th-18th / 16th-17th
Environmental Sci. & Eng.	15th-16th / 14th-15th
Industrial & Systems Eng.	16th-17th / 15th-16th
Materials Sci. & Eng.	16th-17th / 11th-12th
Mechanical Eng.	15th-16th / 14th-15th
Mining Eng.	15th-16th / 15th-16th

the large number of students from 13 different departments in the College of Engineering. To better understand the characteristics of this category, we divided the documents in the College of Engineering based on their corresponding departments. Details are given in Table 8. Thus, we expect document pairs within this college to be more diverse compared to others. From Table 8, we observe that documents from most departments in the College of Engineering have a similar readability consensus between academic abstracts and general audience abstracts. The only outlier is the Department of Computer Science, which shows that general audience abstracts are more difficult to understand than academic abstracts. On the other hand, the Department of Electrical and Computer Engineering alone

represents 15.2% of the total dataset, while maintaining an above-average readability consensus for its general audience abstracts, thus contributing to the higher readability consensus for the College of Engineering. We conjecture as follows: 1) Various complex terminologies from some research fields are inevitably retained in general audience abstracts because they cannot be replaced by simple terms. 2) Concepts in certain areas of engineering research are difficult to explain or rewrite into short and concise sentences. 3) Some engineering students are not as good at simplification writing as students from other colleges.

C Parameter Settings

For the hyperparameters and configuration of DSPT5, we implement DSPT5 in PyTorch and optimize it with the AdamW optimizer. For pre-trained LLMs, we retrieve the generated output through their API by giving the prompt "Generate another version of the provided document for the general audience". We use grid search to tune the hyperparameters. The learning rate is $\in \{5e-2, 5e-3, 5e-4, 5e-5\}$. The contrastive loss weight λ in Equ. 8 and the alignment function weight γ in Equ. 10 are chosen from 0.1 to 0.9 with a step size of 0.2. The number of candidates C_n in Sec. 4.3 is chosen from $\{4, 8, 16\}$. The batch size is 4. The length of the source text, fixed prompt template, and keywords are 512, 16, and 16, respectively. The fixed prompt template used for baselines is: 'Generate another version of the provided document for general audiences.'. We choose the hyperparameters based on the validation set. Because the dataset is not a balanced dataset, where College of Engineering takes up most of the datasets. We split a validation set in the domain of College of Engineering documents. This is because as shown in Table 1, it is not realistic to have a validation set in the Business domain, which has only 63 documents in total. The size of the dataset will only grow as new master's theses or PhD dissertations are submitted. In the future, when the dataset is large enough, we will split out a validation set that includes data from all colleges. The final learning rate we used is 5e-5. However, this can be changed according to different batch sizes when fine-tuning the model. The parameters we used are searched by a validation set from the engineering college documents (which takes the highest percentage of the total documents). The contrastive loss weight is 0.3, the alignment function weight is 0.1, and the number of candidates is 16. If people want to work in a different domain (other than the College of Engineering), different parameters may have different performances. For the temperature in contrastive loss, we use the InfoNCE loss from (van den Oord et al., 2018), and we use the default temperature value of 0.1. The temperature in decoding is 0.5. For model details shown in Figure 1, we use the same T5-encoder that shares parameters to keep our model DSPT5 as small (efficient) as possible. Here are the exact versions for LLM used: ChatGPT: gpt-3.5-turbo-0613 Claude2: Claude2.1 LLaMA: LLaMA2-7b BART: bart-large-cnn FlanT5: flan-t5-base T5: t5-base.

For the ablation study setup, the model is trained on the entire training dataset without sampling. We only do the sampling on the evaluation dataset for the ablation study. Since the data split is 0.8:0.2 on each college domain, the evaluation set is also not a balanced dataset. In total, there are about 988 documents in the evaluation set, which is used for the evaluation in Table 2. After sampling, we have around 120 documents (which is now a balanced evaluation set covering all colleges) for the evaluation set ONLY used for the ablation study Table 3. We reported the average performance including all colleges in Table 3 for the ablation study. Since the College of Engineering takes up almost half of the data shown in Table 1, the result of the average performance will be heavily influenced by the performance of the College of Engineering. Therefore, we sampled a balanced evaluation set to report the ablation study.

D More Results

In this section, we provide the combined test set results of all eight colleges, which is shown in Tabel 9. From Table 9, we observe that our proposed DSPT5 outperforms all other baselines in the evaluation metric of s-BLEU, d-BLEU, BERTScore, BLONDE, ROUGE and METEOR. The closed-source model ChatGPT shows better performance in system-level evaluation metrics such as COMET. We conjecture that this is probably because such pre-trained LLMs (i.e. ChatGPT, Claude2, LLaMA2) are pre-trained on large quantities of documents, which makes these models generate text more fluently, resulting in better system-level evaluation metrics. Alternatively, our proposed model DSPT5 is fine-tuned on our cus-

tomized dataset VTechAGP, which results in better performance in word-level evaluation metrics.

E Case Study

Table 10 shows an example of general audience abstracts generated by different LLMs on VTechAGP (Department of Chemical Engineering). The first few sentences of the document are displayed. Fine-tuned T5 and FLAN-T5 show similar outputs with source text, so we have not included them in Table 10. The key idea of the source text is: Polymers are important in life and its manufacturing process is critical to industry. We observe that the pre-trained LLMs can improve the readability to some extent. For example, ChatGPT and LLaMA2 rewrite 'automotive industry' as 'cars'. Claude2 translates 'Ziegler-Natta catalysts' to 'a specific type of catalyst'. However, some academic domain-specific words (technical terminology) still exist. For example, the generated outputs still contain the phrases 'high-density polyethylene' and 'linear low-density polyethylene' from Claude2 and ChatGPT, 'Ziegler-Natta catalysts' from ChatGPT and LLaMA2. Despite the catalyst terminology, LLaMA2 summarizes the academic text best, which is consistent with the experimental results of the simplicity metric (SARI) in Table 2. In contrast, the fine-tuned generative language models (i.e., BART and Ours) do not introduce these technical terminologies. Fine-tuned BART generates some unrelated phrases (i.e. kinetic parameters), whereas our proposed model DSPT5 can still express the key idea from the source text that polymers are important and widely used in life and their manufacturing process is critical to industry. At the same time, there is no technical terminology in the output text generated by DSPT5. We did not include the output of fine-tuned T5 and FLAN-T5 as they yield similar outputs to the source text. We speculate that this is because there is no appropriate domain-specific prompt provided for fine-tuning T5 and FLAN-T5 with the limited training data.

F Human Evaluation

To provide a comprehensive assessment of the generated general-audience abstract, we conducted a human evaluation involving our proposed model DSPT5 and all other baseline models using three independent experts [12]. Specifically, following a similar setting as (Liu et al., 2024; Li et al., 2024; Song et al., 2024; Kew et al., 2023; Devaraj et al., 2021), our evaluation uses a random sample of 20 abstracts (10 in Computer Science and 10 in all other fields) from the test split VTechAGP considering the workload. Judges are presented with both the academic abstract and generated general-audience abstracts from seven models (DSPT5 and six baselines) for each data sample in a total of 140 abstracts. Using a 1-5 Likert scale, the judges are asked to rate the model output based on five criteria: comprehensiveness, layness, meaning preservation, conciseness, and fluency. Details are shown in guidelines, which clarify the meanings for each criterion in Figure 3 in the Appendix. Table 4 presents the average ratings from our human evaluation.

G Methodology Explanation Details

In Sec. 4, we discuss the contrastive loss infoNCE derived from different learning representations. Here we provide a detailed explanation. To the textual level, r represents a set of keywords (here a set of keywords are textual keywords, and then r is the representation (embedding) of the several textual keywords). As we said, r^{pos} and r^{neg} are not sets of vectors. Instead, r^{pos} is a vector and r^{neg} is a vector. They are the vector representation of a set of textual keywords. If there are m keywords in the golden label and keyBERT returns n keywords (where $m < n$), we sort the keywords by confidence and select the top m keywords. Then we concatenate the words for dot production. For example, if there are two terms in the label, we select two positive keywords from KeyBERT. The selection is based on the ranking of the confidence value provided by KeyBERT. We use T5 encoder, the inputs_embeds are in [batch_size, sequence_length, hidden_size], we concatenate embeddings in the sequence_length dimension. Although dot production is sensitive to the order of the keywords, currently we do a simple concatenation and we leave the works for exploring new ranking approaches to make the keyword extraction module extract closer or more relevant words to the golden label in the future as the main contributions for this paper is about the new dataset VTechAGP and implementations for benchmark models.

[12] All judges have experience in scientific research and hold bachelor, master, and doctorate degree respectively.

Table 9: Combined test results of all eight colleges.

	s-BLEU	d-CLEU	BERTScore	BLONDE	ROUGE1	ROUGE2	METEOR	COMET	SARI	LTCR	FRES	Toxicity
BART	11.24	25.09	84.28	16.82	52.33	27.15	40.58	80.27	36.79	59.19	31.12	0.19
T5	9.75	25.16	84.15	17.20	50.63	26.30	39.62	77.71	37.56	57.44	31.73	0.24
FLAN-T5	9.77	19.93	82.64	16.26	46.57	21.23	35.24	77.66	36.65	63.99	32.98	0.28
Claude2	0.88	2.50	78.28	7.03	26.44	4.57	18.04	70.78	29.62	**70.16**	19.57	0.23
ChatGPT	4.70	14.36	83.77	17.65	48.45	18.52	38.07	**83.84**	36.86	59.40	**35.33**	**0.08**
LLaMA2	3.21	15.71	82.58	14.70	40.43	22.83	41.49	81.54	**44.51**	58.78	32.50	0.29
Ours	**12.41**	**27.02**	**84.75**	**21.11**	**52.71**	**28.59**	**42.15**	80.21	38.01	56.50	32.32	0.23

H Ethical Statements

The VTechAGP dataset, derived from theses and dissertations available in VTechWorks,[13] is shared with a clear understanding of the ethical implications. Virginia Tech, as the institution where these ETDs were submitted, holds the right to distribute the content within the bounds of academic sharing and research purposes. The university's policy ensures that the distribution of such academic materials aligns with educational objectives and respects the authors' intellectual contributions.

Descriptive metadata, including titles, degree types, and departmental affiliations, is generally not subject to copyright restrictions. This metadata is typically viewed as factual information, which falls outside the scope of copyright protection. This understanding allows for the broad dissemination and analysis of metadata in academic and research contexts without infringing upon copyright laws. The abstracts, however, may present a different scenario. While the university facilitates their distribution through VTechWorks, the copyright of the abstract text itself could potentially remain with the authors. Nevertheless, the use of abstracts in the VTechAGP can be justified under the principles of fair use, particularly given the dataset's purpose in advancing scholarly research, education, and understanding. The dataset does not replicate the full ETDs but instead provides a means to study and analyze the structure and nature of academic abstracts themselves, thereby contributing to the field of information retrieval and digital libraries.

[13] Virginia Tech Electronic Theses and Dissertations: https://hdl.handle.net/10919/5534

Table 10: Examples of academic to general audience text paraphrasing comparison. The target text shows the gold label.

Source Text	Polyolefins are one of the most widely used commodity polymers with applications in films, packaging and automotive industry. The modeling of polymerization processes producing polyolefins, including high-density polyethylene (HDPE), polypropylene (PP), and linear low-density polyethylene (LLDPE) using Ziegler-Natta catalysts with multiple active sites, is a complex and challenging task...
Target Text	Almost everything we see around us from furniture, electronics to bottles, cars, etc. are made fully or partially from plastic polymers. The two most popular polymers which comprise almost two-thirds of polymer production globally are polyethylene (PE) and polypropylene (PP), collectively known as polyolefins. Hence, the optimization of polyolefin manufacturing processes with the aid of simulation models is critical and profitable for chemical industry.
BART	Polyolefins are one of the most widely used commodity polymers with applications in films, packaging and automotive industry. The modeling of polymerization processes producing polyfins involves many processes and their kinetic parameters that have been validated by scientific institutions and market models...
Claude2	Polyolefins are a category of plastics that are utilized across many industries for products such as packaging materials, plastic bags, and automobile parts. Modeling the manufacturing processes that create popular polyolefins like high-density polyethylene, polypropylene, and linear low-density polyethylene, when using a specific type of catalyst, involves intricate and difficult steps...
ChatGPT	Polyolefins are commonly used plastics found in items like films, packaging materials, and parts for cars. Creating these plastics, such as high-density polyethylene (HDPE), polypropylene (PP), and linear low-density polyethylene (LLDPE), through polymerization processes involving Ziegler-Natta catalysts with multiple active sites is a complex and challenging undertaking.
LLaMA2	Polyolefins are common plastics used in packaging, films, and car parts. Modeling how these materials are made is complex and challenging, especially when using Ziegler-Natta catalysts with multiple active sites...
Ours	The global population continues to grow, and with it, the demand for polyolefin will further increase. Thus, there will never be a non-polymerization process, and thus, many parts of the global production chain must be produced synthetically. Polyolefin is the most widely used commodity in the automotive and aerospace industries. The manufacturing of polyolefins is challenging since...

Human Evaluation Guideline

This guideline is intended to give annotators a clear understanding of the task and requirements before manual evaluation. Be sure to read the following content carefully.

This task is used to assess the quality of general-audience abstract generated by different models. You are required to complete 20 subtasks in total (10 tasks in all fields and 10 tasks in Computer Science field), each of which will provide you with the original academic abstract and 7 general-audience abstracts generated by different models. You need to score each generated abstracts based on five evaluation indicators, with score of 1 represents the worst and 5 represents the bests. The five evaluation indicators are:

1. Comprehensiveness/Completeness: to what extent does the model output contain all information that might be necessary for a non-expert to understand the high-level topic of the article and significance of the research.
2. Layness: to what extent is the content of the model output comprehensible (or readable) to a non-expert, in terms of both structure and language.
3. Factuality/Correctness/Meaning Preservation: to what extent is the model output factually consistent and grammatically correct.
4. Conciseness/Non-Redundancy: to what extent does the model output contain non-redundant or non-repeated information.
5. Fluency/Coherence: to what extent does the model output read smoothly and naturally, without grammatical, spelling or formatting errors. Sentences should be coherent and consistent with natural reading habits, rather than simply stacking sentences together. Both quality of individual sentence and relationships between sentences should be considered.

Please note that you will not know the seven abstracts is generated by which model respectively, and their order in different subtasks is random.

Evaluation results are only used for this study. All the information will be anonymized, and your personal preferences will not be disclosed. You do not have to bear any responsibility for the risk caused by your evaluation results.

Figure 3: Human evaluation guideline.