Generating Academic Abstracts: Controlled Text Generation Using Metrics from a Target Text Allows for Transparency in the Absence of Specialized Knowledge

Elena Callegari[1,2,*], Peter Vajdecka[3] and Desara Xhura[2]

[1] *University of Iceland, Sæmundargata 2, 102 Reykjavík, Iceland*
[2] *SageWrite ehf., Grettisgata 55, 101 Reykjavík, Iceland*
[3] *Prague University of Economics and Business, nám. W. Churchilla 1938/4 130 67 Praha 3, Czechia*

Abstract

The lack of specialized linguistic knowledge in end users can make it difficult to develop more transparent natural language generation applications. This paper introduces a novel approach to controlled text generation, with a particular emphasis on controlling the stylistic properties of the generated text. By extracting and concatenating numerical metrics—representing various stylistic properties—from a reference text crafted by the target author into the text input of our generation model, we enhance the stylistic alignment between generated and target texts. Our proposed method successfully improves this alignment, surpassing the baseline, and represents a promising first step towards striking a balance between explainable AI and lack of specialized knowledge.

Keywords

Natural Language Generation, Controllable Text Generation, Rule-based Generation, Explainable AI, Academic Abstracts, Stylistic Metrics

1. Introduction

Artificial Intelligence (AI) and Machine Learning (ML) have achieved remarkable success across various practical applications such as natural language processing, face recognition, autonomous driving, image classification, automated clinical diagnosis, and more [1, 2]. Deep learning approaches have even surpassed human performance in numerous domains [1, 3]. However, a significant drawback of deep learning methods is their inherent lack of interpretability [4, 5], which makes them opaque “black boxes". This opacity poses substantial challenges, particularly when attempting to interpret predictive results that might later be found incorrect [5, 6, 7]. Consequently, there is an urgent necessity to add transparency, comprehensibility, and explainability to the outcomes and decisions made by machine learning systems [5, 8, 9, 10].

The need for increased transparency can also be appreciated in the domain of Natural Language Generation (NLG). NLG is a subfield of artificial intelligence that focuses on generating

GENERAL'23: GENerative, Explainable and Reasonable Artificial Learning Workshop 2023, held in conjunction with CHITALY 2023

[*] Corresponding author.

ecallegari@hi.is (E. Callegari); vajp02@vse.cz (P. Vajdecka); desara@sagewrite.com (D. Xhura)

human-like text from structured data or other inputs. Most studies at the interface between Explainable AI (XAI) and NLG have focused on using NLG to generate those additional explanations needed to make a given AI application more transparent [11, 12, 13, 14, 15, 16, 17, 18]. An example is [12], who use NLG to generate text that justifies and explains the rationale for the different classification decisions made in a deep visual recognition task. However, text generation *for the sake* of text generation (e.g. for question-answering, or to automatically generate the summary of an article) could also benefit from additional transparency [19, 20], particularly with the arrival of large, pre-trained models.

The advent of large language models (LLMs) such as GPT [21], T5 [22] and BART [23] has made it possible to generate text that is more diverse, sounds more natural and is more relevant. However, LLMs are essentially still black boxes, lacking interpretability: LLMs always generate text according to the latent representation of the context, making it difficult to control the generation. This has led to the rapid rise of Controlled Text Generation (CTG) studies with transformer-based LLMs. Various methodologies have emerged in the last 3-4 years, with topic and sentiment control being particularly popular areas of application of CTG [24]

2. Controlling Style in NLG: User Knowledge

Despite the impressive progress in NLG, the challenge of producing text outputs that conform to specific writing styles or stylistic preferences persists. Efforts in this direction often employ costly methods such as reinforcement learning [25], or oversimplify the style transfer issue by framing style as a classification problem, categorizing writing styles into specific classes, such as Formal, Poetic, or Topical [26]. However, writing styles encompass a uniqueness and complexity that extends beyond simple categorical classes. Research in stylometry [27, 28, 29, 30, 31] demonstrates that specific writing features, such as punctuation use patterns [28], can help distinguish whether a text was written by a particular individual, underscoring writing style as a distinctive attribute of the author, which cannot be easily constrained into a particular style classification dilemma.

Unless deliberate measures are employed to govern text generation, the resulting style remains implicitly influenced by the training data [32], potentially leading to unnatural and monotonous-sounding text that often significantly deviates from an intended target text, or from the intentions of the end user. Hence, it becomes desirable to have explicit control over various stylistic aspects of text generation. For instance, an author aiming to generate text might want the generation to reflect their aversion to the use of an excessive number of adjectives, or their preference for sentences featuring at most two embedded clauses.

Being able to control generation down to these fine-grained parameters would improve user satisfaction with respect to the outputs of a text-generation model: if users could define and modify fine-grained parameters such as “max. number of adjectives per nominal phrase" and “max number of embedded clauses per sentence" according to their preferences, they would be able to generate NLG outputs that are truly tailored to their individual style. Controlled text generation would also enhance the transparency of deep-learning models: by controlling generation through different linguistic parameters, users would be in a better position to understand the decision-making process behind the generated text. Moreover, by examining these parameters, users could gain insights into how different linguistic parameters are utilized to produce specific text elements. For example, modifying and observing the impact of specific linguistic parameters would enable users to trace how alterations influence the text output, providing an observable cause-effect relationship in the text-generation process.

The main catch preventing us from implementing such fine-grained parameters and allowing the user the possibility to control them is the fact that a significant portion of the population lacks the necessary linguistic background to comprehend and adjust them. Users might not know what the difference between *sentence* and *clause* is, or might struggle to remember what an adjective is. Moreover, knowledge of linguistic terms does not guarantee an ability to deconstruct one style's into specific sub-parameters: we can easily tell whether we like, dislike or are indifferent to the style of a given paper, but determining whether that is due to particular lexical, syntactic or punctuation choices is considerably harder to do.

In this study, a novel method is presented to incorporate stylistic metrics into text generation control. The objective is to achieve text outputs that closely match an intended style. To achieve this, we extract a set of stylistic metrics, represented as numerical values with decimal points, from a reference text written by a target author. These metrics are then incorporated into the input provided to our text-generation model.

The idea behind this methodology is to render NLG more transparent and customizable *without* the need for the user to have any specialized linguistic knowledge to fine-tune or understand the parameters behind the generation. Instead of letting the user set fine-grained parameters to control text generation, we extract them automatically from a reference text that is representative of the style of the target author, and incorporate these parameters into the input of our text-generation model. We believe ours to be a promising approach towards improving and democratizing the explainability of NLG, as no specialized linguistic knowledge is required from the user, who is nonetheless still able to control the output of the generation.

3. Existing Work

We test our approach for controlling text generation on the task of automatically generating academic abstracts using the article as input. Our goal is to produce abstracts that harmonize with the overall style of the article, creating a seamless and coherent extension of the paper itself.

Abstract generation can be seen as a type of summarization problem, a challenge that can be approached using a variety of techniques [33]. Extractive summarization involves identifying key sentences or fragments in the original text and piecing them together to form a summary [33]. In contrast, abstractive summarization or generative summarization aims to generate novel sentences, potentially using new phrasing or condensation, to provide an overview of the contents of the original text [33]. A hybrid approach can combine both techniques, leveraging their respective strengths [34]. Additionally, some researchers have proposed citation-based summarization, where the content of citations is used to help produce the summary [35]. The length and complexity of scientific articles have historically made it difficult to train abstractive summarization models only. There is a distinct lack of research on the generation of abstracts from scientific articles using abstractive summarization techniques directly.

Concerning existing studies that have attempted to control the style of an automatically generated text, important mentions are Syed et al. (2020) ([36]), who control for stylistic properties by fine-tuning on a target author's corpus using denoising autoencoder loss, Wang et al. 2019 ([37]), who incorporate GPT-2 with a rule-based system for formality style transfer, and Singh et al. (2020) ([38]), who, using reinforcement learning, attempt to induce certain lexical target-author attributes by incorporating continuous multi-dimensional lexical preferences of the target author into the language model.

To our knowledge, no one has yet attempted to incorporate text together with decimal numbers as the input of LLM-based summarization systems to enhance the quality of the final summary in the form of an abstract. This current study seeks to bridge this gap by investigating the application and performance of such an approach.

4. Our Approach

4.1. Dataset

As our initial dataset, we decided to use the Huggingface ArXiv dataset[1], which features 215,913 scientific articles along with their respective abstracts. We then eliminated the bibliography section from each article. Next, we computed the word-piece token length [39] and word length for every article. All articles exceeding 2800 words or 3650 word-piece tokens were excluded from the dataset. This step was necessary to adhere to the maximum token length allowed as input, considering the limitations of our Nvidia H100 GPUs, which are the GPUs we used for training.

The resulting dataset consists of 18,175 scientific articles and their corresponding abstract, and can be accessed here[2]. To develop and evaluate an unbiased generative model, we divided this dataset using a 60:20:20 split.

4.2. Stylistic Metrics

Academic authors bring their own distinctive writing styles, which can be influenced by their educational training, research experiences, and personal writing preferences. These variations in style become apparent in multiple aspects, encompassing sentence organization, word selection, formality levels, use and frequency of specialized terminology, and punctuation preferences. The diversity of academic writing styles is made even more significant because of disciplinary differences, as each research field may adhere to specific conventions and norms concerning writing approaches.

In order to replicate a particular author's style, one should ideally account for all these factors. However, in this paper, we only attempt to control the parameters that are outlined below:

1. Average, mean, max, min number of words per sentence, and st. dev. value;
2. Average, mean, max, min word length, and st. dev. value;
3. Average, mean, max, min paragraph length, and st. dev. value;

[1] https://huggingface.co/datasets/scientific_papers

[2] https://www.kaggle.com/datasets/desaraxhura/arxiv-dataset-enhanced-with-stylometric-features

4. Frequencies of different PoS categories (nouns, verbs, adverbs, adjectives, articles);
5. Presence or absence of Oxford commas;
6. Lexical diversity;
7. Number of colons and semicolons for every 500 words;
8. Percentage of words that appear in the 2000-most-frequent English words list;
9. Proportion of sentences containing a subordinate clause;
10. Max number of subordinates per sentence;
11. Proportion of passive verbs over the total number of verbs.

Although these metrics cover just a portion of all conceivable dimensions that could be analyzed, they present a balanced combination of syntactic features (e.g., *maximum number of subordinates*), morphological aspects (*average word length*), lexical characteristics (*lexical diversity*), and purely stylistic elements (*use of Oxford commas*).

We extracted these metrics from the texts in our dataset, resulting in stylistic reports similar to that illustrated in Figure 1. As can be seen in Fig. 1, these reports contain not only text but also numerical values with decimal points.

4.3. Model Selection

After considering several large language models, we ultimately opted for T5 due to its exceptional performance on numerical reasoning tasks [40] and proficiency in numeracy tasks with integer numbers [41]. Nevertheless, the original T5 model was not specifically trained to handle inputs containing both text and numbers with decimal points [22]. To overcome this limitation, we explored the possibility of using Flan T5, a modified version of T5 known for its good performance in few-shot learning [42]. We ran multiple experiments and observed that Flan T5 outperforms the original T5 base model on various summarization tasks, a similar conclusion to the one reached by Chung et al. (2022) ([42]). Based on these observations, we opted for further fine-tuning Flan T5 instead of T5.

4.4. Logic & Experiments

Our aim was to investigate whether and how the inclusion of stylistic reports, computed from various types of target texts (either the original abstract or the full article text), impacted our model's performance.

Our approach consisted of three phases. In the first phase, we trained the model using raw article text as input, without incorporating any stylistic report. This served as our baseline, enabling us to evaluate the model's performance without the influence of stylistic metrics. The model solely relied on the patterns within the input text for making inferences.

Having established the baseline, we proceeded to the second phase of our approach, where we integrated stylistic reports calculated from the original abstract of each paper. The stylistic report and the raw text of each article were combined to create a new input for the model.

In the last phase of the experiment, we built upon the second phase by employing stylistic reports calculated from the raw text of each article (i.e., the article text without the abstract and bibliography), instead of relying on the original abstract. Our hypothesis was that incorporating

Table 1
Stylistic Report - Example

Stylistic metric	Value
Minimum words per sentence:	4
Maximum words per sentence:	35
Average words per sentence:	3.63157896
Mean words per sentence:	8.593157896
Standard deviation of words per sentence:	11.7453393254714
NOUN Proportion:	0.3
VERB Proportion:	0.04
DET Proportion:	0.06
ADJ Proportion:	0.05
ADV Proportion:	0.01
AUX Proportion:	0.02
CONJ Proportion:	0.00
Oxford commas:	0
Lexical Diversity Index:	0.263635465564
Average n. of commas per sentence:	0.65756558577
Average n. of semicolons per sentence:	0.0
Predicted n. of commas per 500 words:	36.34337337096
Predicted n. of semicolons per 500 words:	0.0
The proportion of common words in the text is:	0.25748409645312
Average n. of sentences with subordinate clauses:	0.1
Highest number of subordinates in a sentence:	2
Average Word Length:	3.33
Mean Word Length:	3.3
Maximum Word Length:	29
Min Word Length:	1
Standard Deviation of Word Length:	3.04
Passive Verb Proportion:	0.0

the stylistic report from the entire article would offer a more comprehensive representation of the article's style compared to a report derived solely from the original abstract. Just like in the previous phase, we concatenated the stylistic report with the raw article text to create the input for the Flan T5 model.

Table 2
Rouge Metrics

Model	Rouge 1 F-score	Rouge 2 F-score	Rouge L F-score	Rouge 1 P	Rouge 2 P	Rouge L P	Rouge 1 R	Rouge 2 R	Rouge l R
Abstract stylistic report + Flan T5	**0.342**	**0.128**	**0.302**	**0.430**	**0.162**	**0.379**	**0.313**	**0.120**	**0.276**
Article stylistic report + Flan T5	0.333	0.121	0.293	0.416	0.152	0.366	0.307	0.114	0.271
Baseline: Flan T5	0.335	0.123	0.294	0.420	0.155	0.369	0.306	0.115	0.269

4.5. Fine-tuning Flan T5 models

For fine-tuning, we utilized PyTorch as the framework, running the process concurrently on three Nvidia H100 GPUs with 80 gigabytes of GPU memory.

Table 3
Cosine similarity statistics of generated abstracts and original abstracts

Model	min similarity	max similarity	mean similarity	median similarity	std_dev similarity
Abstract stylistic report + Flan T5	0.028	**0.999**	**0.901**	**0.939**	**0.116**
Article stylistic report + Flan T5	0.042	0.999	0.885	0.934	0.138
Baseline: Flan T5	**0.045**	0.999	0.890	0.935	0.129

$$\textbf{stylistic vector} = \begin{bmatrix} 4 & 35 & 3.63157896 & 8.593157896 & \cdots & 29 & 1 & 3.04 & 0.0 \end{bmatrix}$$

Figure 1: Example of stylistic vector

To train our models, we experimented with multiple different training parameters, acknowledging the constant need for optimization. We reached the best results when we opted for 3 epochs, using a learning rate of 1e-5, a batch size of 3, and the Adam optimizer [43]. We set the maximum input sequence length to 4,000 word-piece tokens, with the stylistic report taking up 350 tokens, allowing for a maximum of 3650 tokens for the article input. For generated abstracts, the maximum output sequence length was set to 400 word-piece tokens. To promote diversity and exploration during training, we employed a sampling parameter set to True. To ensure reproducibility and maintain control over randomization during training, we set the random seed to 42.

Despite our best efforts, the inherent GPU memory limitations constrained us from training on even longer scientific articles, restricting us to papers of up to 4,000 word-piece tokens. Though these articles, at 4,000 word-piece tokens, are on the longer end of what we categorize as “short papers”, this limitation underscores the severity of the imposed hardware constraints.

5. Results

In assessing our experiments, we employed Rouge metrics [44] to assess the generated abstract’s quality by examining word overlap with the original abstract. The outcomes are presented in Table 2. The inclusion of a stylistic report calculated from the original abstract results in an increased overlap compared to the baseline, as evidenced by all Rouge metrics. However, integrating a stylistic report calculated from the article text does not lead to any improvement over the baseline model’s results.

We also conducted cosine similarity calculations between the numerical values present in the stylistic reports of the generated abstracts and those derived from the original abstracts. This allowed us to determine the effectiveness of incorporating a stylistic report into the Flan T5 model as a CTG technique, aligning the stylistic attributes of the generated abstracts with those of the originals.

To compute the cosine similarity, we transformed each stylistic report (illustrated in Fig. 1) into a vector representation (demonstrated in Fig. 1). Subsequently, we computed the cosine similarity between the stylistic vectors of the generated abstracts and those of the original abstracts. In Table 3, we present the cosine similarity statistics obtained from the test dataset, with the mean and standard deviation being the most relevant values. Notably, the model that incorporates stylistic reports from the original abstracts produces abstracts that are most stylistically similar to the original ones. Furthermore, this model achieves the lowest variability around the mean, indicated by the standard deviation, ensuring that the stylistic similarities are the least divergent compared to the other models.

6. Conclusions

Explainable AI is crucial for fostering trust, transparency, and interpretability of NLG systems. Rule-based control in NLG provides a practical approach to increase explainability by leveraging explicit rules to govern the text-generation process. However, most potential end users of NLG applications may lack the specialized linguistic knowledge and introspection needed to control text generation down to truly fine-grained linguistic parameters.

In this paper, we have introduced a novel approach to control the stylistic features of text generation by incorporating stylistic metrics, featuring decimal numbers, from a target text into Transformer-based language models. Our results demonstrate the potential of using stylistic metrics extracted from the original abstract as control mechanisms for abstract generation; this method allowed us to achieve better stylistic alignment with the target text than what we achieved with the baseline alone. In the future, we would like to experiment with vectors that include more metrics than the 11 dimensions we included in this first study, to see how much alignment improves when additional linguistic dimensions are included in the stylistic report that is used as the input for our text-generation model.

While we were successful in improving alignment when using the original abstract as target text, we observed a slight decrease in Rouge and vector similarity scores when calculating stylistic metrics on the full article instead. Several factors might be responsible for this difference. Firstly, the presence of multiple authors in STEM articles (articles in the ArXiv dataset are all STEM articles) might mean that different sections of an article are written by different individuals, resulting in a multitude of different styles being used throughout the paper. To explore this further, it would be interesting to conduct a study solely using single-authored articles as input text, to examine whether this nullifies the observed difference between using the original abstract and the full article text to calculate the stylistic report.

Secondly, it is possible that different sections of an article require distinct writing styles. For example, the *Methodology* section might differ in style from the *Introduction* or *Conclusions* sections. Investigating variations in writing styles across different sections and exploring stylistic patterns that better capture each section could be a valuable avenue for future research.

Overall, our findings suggest that the introduced CTG technique has the potential to assist researchers in refining specific sections of an article to align them with their desired writing style. Additionally, our methods offer the flexibility to finely adjust certain sections to match a target writing style, providing researchers with greater control over the stylistic properties of their generated text. Finally, our proposed CTG technique offers an effective way of enhancing the transparency of NLG applications without the need for end users to have any specialized linguistic knowledge, as the fine-grained parameters that are used to control the text generation are extracted automatically from a reference text that the end user can pick themselves.

A limitation of this study is the lack of a qualitative human evaluation of the generated abstracts and their alignment with the original abstract. We have not resorted to human evaluation due to the high-level expertise required to evaluate abstracts of highly technical papers. This is an often-cited issue in NLG studies [36, 38]. Despite the lack of such evaluation, our results are promising and offer a plausible line of research for controlling text generation using a target text to extract the control parameters.

7. Acknowledgements

This study was partly supported by a grant from Rannís, the Icelandic Institute for Research, and a grant by the European Union (Women TechEU, European Innovation Ecosystems programme, Horizon Europe). Computational resources were provided by the e-INFRA CZ project (ID:90254), supported by the Ministry of Education, Youth and Sports of the Czech Republic.

References

[1] A. Adke, S. Ghorpade, R. Chaudhari, S. Patil, Navigating the confluence of machine learning with deep learning: Unveiling cnns, layer configurations, activation functions, and real-world utilizations (2023).

[2] H. Wang, T. Fu, Y. Du, W. Gao, K. Huang, Z. Liu, P. Chandak, S. Liu, P. Van Katwyk, A. Deac, et al., Scientific discovery in the age of artificial intelligence, Nature 620 (2023) 47–60.

[3] S. Ahmad, I. Shakeel, S. Mehfuz, J. Ahmad, Deep learning models for cloud, edge, fog, and iot computing paradigms: Survey, recent advances, and future directions, Computer Science Review 49 (2023) 100568.

[4] A. Ross, F. Doshi-Velez, Improving the adversarial robustness and interpretability of deep neural networks by regularizing their input gradients, in: Proceedings of the AAAI conference on artificial intelligence, volume 32, 2018.

[5] C. Rudin, Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead, Nature machine intelligence 1 (2019) 206–215.

[6] C. Rudin, Why black box machine learning should be avoided for high-stakes decisions, in brief, Nature Reviews Methods Primers 2 (2022) 81.

[7] J. Petch, S. Di, W. Nelson, Opening the black box: the promise and limitations of explainable machine learning in cardiology, Canadian Journal of Cardiology 38 (2022) 204–213.

[8] A. Holzinger, B. Malle, P. Kieseberg, P. M. Roth, H. Müller, R. Reihs, K. Zatloukal, Towards the augmented pathologist: Challenges of explainable-ai in digital pathology, arXiv preprint arXiv:1712.06657 (2017).

[9] A. Holzinger, M. Plass, K. Holzinger, G. C. Crisan, C.-M. Pintea, V. Palade, A glass-box interactive machine learning approach for solving np-hard problems with the human-in-the-loop, arXiv preprint arXiv:1708.01104 (2017).

[10] A. McGovern, R. Lagerquist, D. John Gagne, G. E. Jergensen, K. L. Elmore, C. R. Homeyer, T. Smith, Making the black box more transparent: Understanding the physical implications of machine learning, Bulletin of the American Meteorological Society 100 (2019) 2175–2199.

[11] J. M. Alonso, A. Ramos-Soto, E. Reiter, K. van Deemter, An exploratory study on the

benefits of using natural language for explaining fuzzy rule-based systems, in: 2017 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE), IEEE, 2017, pp. 1–6.

[12] L. A. Hendricks, Z. Akata, M. Rohrbach, J. Donahue, B. Schiele, T. Darrell, Generating visual explanations, in: Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part IV 14, Springer, 2016, pp. 3–19.

[13] S. Anjomshoae, K. Främling, A. Najjar, Explanations of black-box model predictions by contextual importance and utility, in: Explainable, Transparent Autonomous Agents and Multi-Agent Systems: First International Workshop, EXTRAAMAS 2019, Montreal, QC, Canada, May 13–14, 2019, Revised Selected Papers 1, Springer, 2019, pp. 95–109.

[14] E. Reiter, Natural language generation challenges for explainable ai, arXiv preprint arXiv:1911.08794 (2019).

[15] J. M. Alonso, S. Barro, A. Bugarín, K. van Deemter, C. Gardent, A. Gatt, E. Reiter, C. Sierra, M. Theune, N. Tintarev, et al., Interactive natural language technology for explainable artificial intelligence, in: International Workshop on the Foundations of Trustworthy AI Integrating Learning, Optimization and Reasoning, Springer, 2020, pp. 63–70.

[16] E. Mariotti, J. M. Alonso, A. Gatt, Towards harnessing natural language generation to explain black-box models, in: 2nd Workshop on Interactive Natural Language Technology for Explainable Artificial Intelligence, 2020, pp. 22–27.

[17] I. Donadello, M. Dragoni, Bridging signals to natural language explanations with explanation graphs, in: Proceedings of the 2nd Italian Workshop on Explainable Artificial Intelligence, 2021.

[18] S. Kang, B. Chen, S. Yoo, J.-G. Lou, Explainable automated debugging via large language model-driven scientific debugging, arXiv preprint arXiv:2304.02195 (2023).

[19] H. Zhao, H. Chen, F. Yang, N. Liu, H. Deng, H. Cai, S. Wang, D. Yin, M. Du, Explainability for large language models: A survey, arXiv preprint arXiv:2309.01029 (2023).

[20] M. Danilevsky, K. Qian, R. Aharonov, Y. Katsis, B. Kawas, P. Sen, A survey of the state of explainable ai for natural language processing, arXiv preprint arXiv:2010.00711 (2020).

[21] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, I. Sutskever, et al., Language models are unsupervised multitask learners, OpenAI blog 1 (2019) 9.

[22] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, P. J. Liu, Exploring the limits of transfer learning with a unified text-to-text transformer, The Journal of Machine Learning Research 21 (2020) 5485–5551.

[23] M. Lewis, Y. Liu, N. Goyal, M. Ghazvininejad, A. Mohamed, O. Levy, V. Stoyanov, L. Zettlemoyer, Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension, arXiv preprint arXiv:1910.13461 (2019).

[24] H. Zhang, H. Song, S. Li, M. Zhou, D. Song, A survey of controllable text generation using transformer-based pre-trained language models, arXiv preprint arXiv:2201.05337 (2022).

[25] R. Liu, G. Xu, C. Jia, W. Ma, L. Wang, S. Vosoughi, Data boost: Text data augmentation through reinforcement learning guided conditional generation, in: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP), Association for Computational Linguistics, 2020. URL: https://doi.org/10.18653%2Fv1%2F2020.emnlp-main.726.

[26] K. Yang, D. Klein, Fudge: Controlled text generation with future discriminators, ArXiv abs/2104.05218 (2021). URL: https://api.semanticscholar.org/CorpusID:233210709.

[27] E. Stamatatos, A survey of modern authorship attribution methods, Journal of the American Society for information Science and Technology 60 (2009) 538–556.

[28] H. Gómez-Adorno, J.-P. Posadas-Duran, G. Ríos-Toledo, G. Sidorov, G. Sierra, Stylometry-based approach for detecting writing style changes in literary texts, Computación y Sistemas 22 (2018) 47–53.

[29] Y. Sari, M. Stevenson, A. Vlachos, Topic or style? exploring the most useful features for authorship attribution, in: Proceedings of the 27th international conference on computational linguistics, 2018, pp. 343–353.

[30] R. Sarwar, C. Yu, N. Tungare, K. Chitavisutthivong, S. Sriratanawilai, Y. Xu, D. Chow, T. Rakthanmanon, S. Nutanong, An effective and scalable framework for authorship attribution query processing, IEEE Access 6 (2018) 50030–50048.

[31] K. Lagutina, N. Lagutina, E. Boychuk, I. Vorontsova, E. Shliakhtina, O. Belyaeva, I. Paramonov, P. Demidov, A survey on stylometric text features, in: 2019 25th Conference of Open Innovations Association (FRUCT), IEEE, 2019, pp. 184–195.

[32] K.-H. Zeng, M. Shoeybi, M.-Y. Liu, Style example-guided text generation using generative adversarial transformers, arXiv preprint arXiv:2003.00674 (2020).

[33] N. I. Altmami, M. E. B. Menai, Automatic summarization of scientific articles: A survey, Journal of King Saud University-Computer and Information Sciences 34 (2022) 1011–1028.

[34] L. Xiao, L. Wang, H. He, Y. Jin, Copy or rewrite: Hybrid summarization with hierarchical reinforcement learning, in: Proceedings of the AAAI Conference on Artificial Intelligence, volume 34, 2020, pp. 9306–9313.

[35] M. Yasunaga, J. Kasai, R. Zhang, A. R. Fabbri, I. Li, D. Friedman, D. R. Radev, Scisummnet: A large annotated corpus and content-impact models for scientific paper summarization with citation networks, in: Proceedings of the AAAI conference on artificial intelligence, volume 33, 2019, pp. 7386–7393.

[36] B. Syed, G. Verma, B. V. Srinivasan, A. Natarajan, V. Varma, Adapting language models for non-parallel author-stylized rewriting, in: Proceedings of the AAAI Conference on Artificial Intelligence, volume 34, 2020, pp. 9008–9015.

[37] Y. Wang, Y. Wu, L. Mou, Z. Li, W. Chao, Harnessing pre-trained neural networks with rules for formality style transfer, in: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), 2019, pp. 3573–3578.

[38] H. Singh, G. Verma, B. V. Srinivasan, Incorporating stylistic lexical preferences in generative language models, arXiv preprint arXiv:2010.11553 (2020).

[39] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, I. Polosukhin, Attention is all you need, Advances in neural information processing systems 30 (2017).

[40] P.-J. Yang, Y. T. Chen, Y. Chen, D. Cer, Nt5?! training t5 to perform numerical reasoning, arXiv preprint arXiv:2104.07307 (2021).

[41] K. K. Pal, C. Baral, Investigating numeracy learning ability of a text-to-text transfer model, in: Findings of the Association for Computational Linguistics: EMNLP 2021, Association for Computational Linguistics, Punta Cana, Dominican Republic, 2021, pp. 3095–3101. URL: https://aclanthology.org/2021.findings-emnlp.265. doi:10.18653/v1/2021.findings-emnlp.265.

[42] H. W. Chung, L. Hou, S. Longpre, B. Zoph, Y. Tay, W. Fedus, E. Li, X. Wang, M. Dehghani, S. Brahma, et al., Scaling instruction-finetuned language models, arXiv preprint arXiv:2210.11416 (2022).

[43] D. P. Kingma, J. Ba, Adam: A method for stochastic optimization, arXiv preprint arXiv:1412.6980 (2014).

[44] C.-Y. Lin, Rouge: A package for automatic evaluation of summaries, in: Text summarization branches out, 2004, pp. 74–81.