國立臺灣大學電機資訊學院電機工程學系
碩士論文
Department of Electrical Engineering
College of Electrical Engineering and Computer Science
National Taiwan University
Master’s Thesis

停止說‘Delve’！通過復合分佈對齊適配大型語言模型以進行文本風格化

“Stop Saying Delve!” Adapting Large Language Models for Text Stylization with Composite Distributional Alignment

張立憲
Li-Hsien Chang

指導教授: 陳銘憲 博士
Advisor: Ming-Syan Chen, Ph.D.

中華民國 113 年 7 月
July, 2024

doi:10.6342/NTU202403360

國立臺灣大學碩士學位論文

口試委員會審定書

MASTER'S THESIS ACCEPTANCE CERTIFICATE
NATIONAL TAIWAN UNIVERSITY

停止說 'Delve' ！通過復合分佈對齊適配大型語言模型以進行文本風格化

Stop saying Delve!"Adapting Large Language Models for Text Stylization with Composite Distribution Alignment

本論文係___張立憲____(姓名)_ R10921A37__（學號）在國立臺灣大學電機工程學系完成之碩士學位論文，於民國 113 年 07 月 24日承下列考試委員審查通過及口試及格，特此證明。

The undersigned, appointed by the Department of Electrical Engineering on _24__(date)____07 (month)__2024__ have examined a Master's thesis entitled above presented by_ Li-Hsien Chang_ ______________________ (student ID) candidate and hereby certify that it is worthy of acceptance.

口試委員 Oral examination committee:

陳銘憲 帥宏翰 [illegible]

（指導教授 Advisor）

丁川康

系主任 Director: 李建模

Acknowledgements

完成這篇碩士論文，首先要感謝我的指導教授陳銘憲老師，提供實驗室資源，讓我得以嘗試各種想法。老師在過程中給予的建議，幫助我在想法上取得突破。

另外，也要感謝口試委員陳銘憲教授、帥宏翰教授、丁川康教授以及吳齊人教授在百忙中抽空參加我的論文口試會議，並提供了許多創新的想法和建議，讓我受益良多。

特別感謝雯文學長和方睿學長一路以來在研究上的協助，從情境的發現到實驗的突破，都提供了許多精闢的見解。還有感謝津源學長在實驗中孜孜不倦的幫助，這篇論文能順利完成，要感謝學長們不厭其煩地回答我的問題。

除此之外，也要感謝實驗室其他成員的相互支持與陪伴。特別感謝雅惠幫忙處理實驗室的各種事務，還有我的同學：莉亞、朝棋、宥辰、依婷、佳儀、子仙、盈樺、昱宏、沛蓉、怜均，陪伴我度過艱難的碩士生涯。

最後，要感謝我的家人，他們在經濟和精神上給予的支柱，使我得以順利畢業。

doi:10.6342/NTU202403360

摘要

自 ChatGPT 推出以來，使用者——尤其是非英語母語者——使用大型語言模型（LLMs）提供的服務，有效地幫助了他們表達想法和生產內容。然而，近期有觀察到大型語言模型使用率正急劇上升，也暴露了其輸出文本具有可識別的寫作風格，進而讓使用者沒有辦法去有效提高生產力且被污名化。爲了要去處理這個問題，重要的地方在於使大型語言模型所產出的內容與人類撰寫的文本相似。本次的研究中，我們提出複合分佈對齊（Composite Distributional Alignment，簡稱 CoDA），其中包括零階偏差對齊啓發式算法（Zeroth-order Bias Alignment Heuristics，簡稱 ZoBAH）和判別器自舉提名（Discriminator Bootstrapped Nomination，簡稱 DiBoN）。CoDA 通過在 ZoBAH 中以正反文本的方法篩選文本中具有偏差的字符並給予校正，和在 DiBoN 中針對動態分數重新調整大型語言模型產出的字符流程。具體來説，ZoBAH 解決了大型語言模型產出與專家文本之間在詞彙層面的統計差異，而 DiBoN 進一步結合現成的 AI 檢測器、句法和語義特徵，考慮了更廣範圍的文本差異。

我們在 Multi-XScience 和 BAWE 資料集上的實驗，證實了 CoDA 的可行性。在白盒和黑盒場景中，它在傳統的字詞上、句型和文義檢測上取得了大量的改進。與現有最好的標準方法相比，將最先進 AI 檢測器的檢測率在白盒場景中降低了近 20%。另外，這個研究也展示了 CoDA 的可轉移性，展示了其構建通用權重的潛力，有效地消除了字詞、句型和文義特徵層面的誤差。

關鍵字：文字風格對齊、大型語言模型誤差

doi:10.6342/NTU202403360

Abstract

After the release of ChatGPT, large language models (LLMs) have provided significant assistance to non-native English speakers by refining their scientific writings. However, these models often generate text with a distinctive style that could potentially stigmatize its users. To mitigate this effect, it is essential to tailor LLM-generated content to more closely mimic human-produced texts. In this work, we present *Composite Distributional Alignment (CoDA)*, which includes *Zeroth-order Bias Alignment Heuristics (ZoBAH)* and *Discriminator Bootstrapped Nomination (DiBoN)*. CoDA modifies the autoregressive token generation in LLMs by adjusting logits using static biases from in data-driven fashion in ZoBAH and dynamic scores from DiBoN for top token options. Specifically, ZoBAH addresses word-level statistical disparities between LLM outputs and expert texts, while DiBoN further adjusts pattern-level criteria, incorporating off-the-shelf AI detectors, syntactic, and semantic features.

Our extensive tests on both Multi-XScience and BAWE datasets confirm that CoDA significantly outperforms existing methods according to standard word- and pattern-level metrics under both white-box and black-box conditions, achieving up to 15% reduction in detection rates by advanced AI detectors compared with the strongest baseline in white-box setups. Furthermore, our studies reveal that CoDA is effective not only in adjusting biases but also in transferring knowledge across different contexts, thereby improving overall text quality, which suggests its utility in developing a universal weight capable of mitigating biases effectively at multiple levels.

Keywords: Text Style Alignment, LLMs' Bias

doi:10.6342/NTU202403360

Contents

doi:10.6342/NTU202403360

doi:10.6342/NTU202403360

List of Figures

doi:10.6342/NTU202403360

List of Tables

doi:10.6342/NTU202403360

Chapter 1

Introduction

On November 30, the release of ChatGPT [1] have dramatically impacted the realm of natural language processing, demonstrating exceptional capabilities in tasks like summarization [2], machine translation [3], and question answering [4].

Among these applications, the most transformative ability of LLMs is their capacity to convert amateur text into sophisticated, high-quality paragraphs. This democratizes content creation, especially for non-native English-speaking researchers in scientific fields, as most conferences only accept publications in English [5]. By lowering the barrier to academic writing, LLMs enable researchers to focus on their domain-specific work, thereby enhancing the impact and inclusivity of scientific research [6].

Despite the recent impressive performance of LLMs, inconsistent results such as hallucinations, toxic, and biased text are observed when these powerful tools are misused. This issue is particularly prevalent among non-native speakers, leading to a significant reduction in information accuracy across various tasks [7].

Additionally, many commercial AI text classifiers display bias against writers with non-dominant language backgrounds, disproportionately misclassifying their essays as AI-generated [8]. To avoid being marginalized in evaluative or educational contexts, non-native speakers must use LLMs more frequently to enhance their vocabulary and sound more 'native.'

However, subsequent studies have found that LLM-generated text exhibits a distinct, unnatural, and easily identifiable style [9–11]. For instance, [9, 10] presents a simple and effective method using a maximum likelihood approach to predict the proportion of AI-generated text in large corpora. Their findings reveal trends in AI usage in peer reviews, abstracts, and introduction sections at several top conferences. Similarly, [11] observed a sharp increase in the use of words like *delve* and *potential* in 14 million PubMed abstracts

doi:10.6342/NTU202403360

from 2010 to 2024, coinciding with the launch of ChatGPT.

This perception of artificiality can undermine inclusivity, as LLM-assisted work may be judged for its perceived artificiality rather than its content. This can lead to potential stigma for users, particularly non-native speakers, who are more vulnerable when detection models are deployed at scale. This contradicts the vision of LLMs to promote equal access to information globally. [12]. Improving LLM and classifier designs is crucial to mitigate biases and support non-native speakers in creating authentic, high-quality content, fostering inclusivity in academia. However, the stigma and rigid word choices in LLM-generated scientific writing still need to be solved and explored in current research.

The most closely related areas of study focus on text stylization. [13] harnessed the power of LLMs like GPT-2 by using a specific author's works to fine-tune the model, successfully emulating the author's style in both lexical and syntactic dimensions., which is not suitable for current LLMs due to the enormous computational efforts required. [14, 15] leveraged current off-the-shelf LLMs for stylization by employing various prompting strategies without additional training data. Nevertheless, there remains a lack of ability to identify and mitigate inherent biases in LLMs and to steer the generation process towards a more unbiased output. [16–18] proposed "guided strategies" that decouple LLMs from a post-processing module, guiding text generation only during the inference stage. This flexible, plug-and-play approach becomes increasingly advantageous as LLM parameters grow. However, this method cannot be applied to SOTA LLMs because of their black-box nature, and it increases computational overhead with an additional post-processing stage.

In this work, we introduce two novel approaches to mitigate biases in LLMs: *Composite Distributional Alignment (CoDA)*, which includes *Zeroth-order Bias Alignment Heuristics (ZoBAH)* and *Discriminator Bootstrapped Nomination (DiBoN)*. These methods aim to enhance paraphrasing tasks in scientific writing. CoDA modifies the autoregressive token generation in LLMs by adjusting logits using a static bias from ZoBAH and dynamic scores from DiBoN for top token options. Precisely, ZoBAH adjusts word-level statistical disparities between LLM outputs and expert texts, strategically refining word distribution to more closely align with the human corpus. This adjustment involves storing tokens with their corresponding weights and iteratively refining this word-weight distribution in a gradient descent manner, effectively enhancing the likelihood of tokens that match human-preferred words while diminishing those favored in LLM outputs. This alignment process is offline training and only requires **1.5** times more training time compared to the inference time of the best guided-based method.

Concurrently, DiBoN further adjusts pattern-level criteria, including surface features, syntactic features [13], and established AI-text detectors [19], thus achieving a more thorough capture of potential biases with our ensemble discriminator. By selecting possible token

doi:10.6342/NTU202403360

sequences and reranking each token priority based on their corresponding scores, we can effectively steer LLM-generated results toward the distribution of the human corpus.

Finally, we compare our methods against related works, including traditional prompt-based techniques and guided strategies, by conducting experiments on abstract sections from six diverse domains within the arXiv paper dataset [20] and the L1/L2 dataset from the British Academic Written English (BAWE) corpus [21] to validate the abilities for debiasing and text quality enhancement. The outputs produced by our approaches exhibit remarkable debiasing capabilities for both white-box and black-box scenarios, as evidenced by traditional lexical metrics, pattern-based metrics, model-based metrics, and even the most advanced AI-detector models. Beyond this, we extend our investigation into the generalizability of our methods across various domain datasets. Additionally, we conduct a comparative analysis of our model against commercial academic writing assistants, showcasing its versatility and robustness in diverse applications.

Our contributions are summarized below:

- We present our innovative approach, *Composite Distributional Alignment (CoDA)*, which includes *Zeroth-order Bias Alignment Heuristics (ZoBAH)* and *Discriminator Bootstrapped Nomination (DiBoN)*, to effectively mitigate biases in LLM outputs at both the word-level and pattern-level.
- We showcase the effectiveness and versatility of our debiasing methods by rigorously evaluating them across multiple domain datasets and various LLM settings, benchmarking them against decoding-time methods, such as prompt-based and guided-based approaches.

doi:10.6342/NTU202403360

Chapter 2

Related work

2.1 Controllable Text Generation

The advent of deep learning has sparked a surge in research on DL-enabled controllable text generation (CTG). Early approaches utilized deep generative models, such as Generative Adversarial Networks (GANs), Variational Autoencoders (VAEs), and Energy-based Models. However, These DL-based techniques depend extensively on vast datasets, making it challenging to do cross-domain tasks. Concurrently, Large Language Models (LLMs) have emerged as a novel framework in Natural Language Processing (NLP), deriving advantages from their reliance on extensive data and unsupervised learning with the Transformer architecture[22]. Two primary strategies have surfaced to steer their language production capabilities: training modifications and decoding-time interventions.

Regarding methods applicable at decoding time, [16] presented PPLM. In this approach, an attribute-specific model is trained on top of a PLM's final hidden layer to adjust hidden states via gradient backpropagation for controlled output. GEDI, by [17], uses a small class-conditional LLM as a discriminator to steer text generation in larger models like GPT-2 and GPT-3. Lastly, [18] introduced DExperts, which re-ranks outputs from PLMs using opinions from both expert and anti-expert models at the decoding stage to guide content toward desired attributes.

For training approaches, [23] created a class-conditioned pre-trained LLM trained on text with specific attribute control code prefixes.[24] added an attribute control module to the original PLM, enabling fine-grained control over text generation at the word and pattern levels. [25] examined different configurations of style rewards for a reinforcement learning (RL) approach to control the generation in a multi-style fashion.

Unlike these approaches, our work focuses on mitigating inherent writing style biases

doi:10.6342/NTU202403360

in LLMs. By employing a zeroth-order gradient method combined with a guided-based approach, we aim to enhance controllability and reduce bias in text generation.

2.2 Text Style Transfer

The task of Text Style Transfer (TST) is crucial for modulating the stylistic features of the text while retaining its semantic core.[26]. Definitions of text style vary, covering aspects like semantics content[27], levels of toxicity[28], or features related to authorship[29]. Initial techniques mainly concentrated on adapting models to produce text reflecting a specific style. For example, parallel corpora were employed[30] for converting texts into Shakespearean English and vice versa. In contrast, Large Language Models (LLMs) like GPT-2[13] were used for style imitation based on an author's collected works without needing parallel texts.

More recent strategies involve using LLMs combined with prompt engineering to achieve style transformation. A method [31] called augmented zero-shot learning was introduced, which uses a variety of sentence rewriting examples to enable few-shot prompting without relying on task-specific samples. Furthermore, a study[15] on the potential use of pre-trained LLMs for the controlled generation of stylized text through prompts mimicking an author's style was carried out. Despite these innovations, there still needs to be more control and clarity over the generated outputs.

2.3 Bias within LLMs

Large Language Models (LLMs) are known to manifest biases related to a myriad of demographic indicators such as gender, nationality, race, religion, and sexual orientation.[32] Research has highlighted that these models tend to give lesser probability scores to words associated with minority identities, including "ace," "AAPI," "AFAB," and "pagan" [33], alongside neutral or non-binary pronouns like "they" or "xe" [34]. Furthermore, it is observed that LLMs disproportionately underrepresent countries with lower economic outputs in terms of accurate country name predictions[35]. In addition to these issues, troubling associations have been identified where machine learning algorithms more frequently associate negative traits such as greediness with Jewish individuals rather than Christians[36]. Our research aims to identify and correct the subtler biases in word selection and stylistic preferences exhibited by LLMs to foster improved mitigation strategies.

doi:10.6342/NTU202403360

Chapter 3

Problem

3.1 LLMs' basic operation

Based on the transformer architecture [37], LLMs process text as a sequence of tokens. For a sequence of tokens $x = \{x_0, \ldots, x_n\}$, LLMs are trained to calculate the overall probability $p(x)$. This is formally represented as a product of conditional probabilities by recursively applying the chain rule [38, 39].

$$p(x) = \prod_{t=1}^{n} p_{LLM}(x_t|x_{<t}), \tag{3.1}$$

where LLM is defined as the model and $x_{<t}$ indicates the sequence of tokens prior to the tth token $(x_0, \ldots, x_{t-1})$. In particular, upon generating the nth token, LLM computes the logit scores (i.e., a unnormalized values from the transformer head over the whole token vocabulary) $\mathbf{z}_t \in \mathbb{R}^{|\mathcal{V}|}$, where $\mathcal{V}$ is the vocabulary size of LLM. Therefore, the conditional probability distribution for the next token generation of LLM can be written as:

$$p_{LLM}(X_t \mid x_{<t}) = \text{softmax}[\mathbf{z}_t] \tag{3.2}$$

The token x_t will then be sampled from a (softmax) normalized result of the logits $\mathbf{z_t}$ as and the next token is generated by sampling $x_t \sim p_{LLM}(X_t \mid x_{<t})$.

3.2 Distributional Text Revision Style Alignment

Given a reference corpus of document $\mathcal{C}_{\text{ref}} = \{c_i\}_{i=1}^{n}$ and a source corpus $\mathcal{C}_{\text{source}} = \{c_i\}_{i=1}^{n}$, the objective is to revise a source text x using LLMs to ensure the token generation process aligns distributionally with $\mathcal{C}_{\text{ref}}$. Instruction prompts x_{instruct}, such as "Please revise the

doi:10.6342/NTU202403360

following text," may precede the input text to guide the LLMs in the revision process.[1] The LLM then generates a revised text $\tilde{x} = LLM(x, x_{\text{instruct}})$ based on the conditional probabilities. This process results in a revised corpus $\mathcal{C}_{\text{revised}} = \{LLM(x, x_{\text{instruct}}) \mid x \in \mathcal{C}_{\text{source}}\}$. The goal is to minimize the distributional distance between $\mathcal{C}_{\text{revised}}$ and $\mathcal{C}_{\text{ref}}$, such that

$$\mathcal{D}(\mathcal{C}_{\text{revised}}, \mathcal{C}_{\text{ref}}) \approx 0, \tag{3.3}$$

where $\mathcal{D}$ represents some metric measuring the distance between two distributions.

Given the LLM's tendency to use words that do not align with human preferences, our motivation is to apply our methods to align the LLM's output distribution more closely with that of expert human-authored texts.

[1] An example of an instruction prompt is "Please revise the following text..."

doi:10.6342/NTU202403360

Chapter 4

Methodology

4.1 Zeroth-order Bias Alignment Heuristics (ZoBAH)

We present *Zeroth-order Bias Alignment Heuristics (ZoBAH)*, a data-driven, decoding-time method to steer both black-box and white-box LLMs to align with a positive corpus (C_h) and away from a negative corpus (C_l) at the token level, whose overview is presented in Figure 4.1. By simulating zeroth-order gradient estimation, we use logit offsets as perturbations to adjust LLM outputs without accessing internal information. This simple yet effective method demonstrates significant debiasing capabilities in offline training, making it ideal for modern LLM usage scenarios.

First, our heuristic motivation is to bridge the gap between C_h and C_l through token substitution. We define two token lists for the human corpus and the LLM corpus, respectively:

$$T_h = \{\text{token} \mid \exists c_i \in C_h, \text{ s.t. token} \in c_i\}, \quad T_l = \{\text{token} \mid \exists c_i' \in C_l, \text{ s.t. token} \in c_i'\} \tag{4.1}$$

Next, we calculate the proportional frequencies for these token lists:

$$P(t) = \frac{\#t \text{ appears in the corpus}}{\text{total \# token in the corpus}} = \frac{\sum_{c \in C} \mathbf{1}\{t \in c\}}{|C|} \tag{4.2}$$

Let $P_h(\text{token})$ and $P_l(\text{token})$ represent the proportional frequencies of a token in the human-authored and LLM-revised corpora, respectively. Both are of size V, where V represents the vocabulary size of the LLM.

From a statistical perspective, to quantify the excessive usage of certain tokens in C_h and C_l, we define the frequency ratio $F(t)$ as an indicator of word-level bias for both C_h and C_l. The frequency ratios are given by:

doi:10.6342/NTU202403360

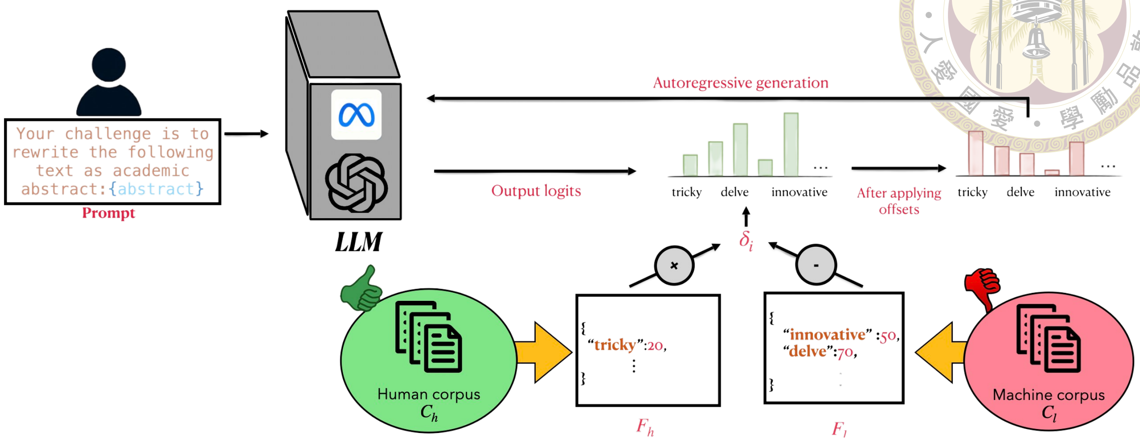


Figure 4.1: Overview of the Zeroth-order Bias Alignment Heuristics (ZoBAH)

$$F(t) = \begin{cases} \frac{P_l(t)}{P_h(t)} & \text{if } \frac{P_l(t)}{P_h(t)} > 1 \text{ and } t \in T_h \cap T_l \\ 0 & \text{otherwise} \end{cases} \tag{4.3}$$

We focus on the tokens used by both C_h and C_l to capture general vocabulary.

According to the design rationale, we hereby provide a detailed exposition of *ZoBAH* in Algo 1. *ZoBAH* consists of three phases.First, for each iteration i, we synthesize C_l and compute F_l and F_h. We then treat F_l and F_h as gradients in the optimization process:

$$gradient = \beta \cdot F'_l[\text{token}] - \alpha \cdot F_h[\text{token}] \tag{4.4}$$

Where β and α are hyperparameters controlling the amount of modification injected into the LLM. Similar to the gradient optimization process, we use a momentum mechanism to record iteration information and obtain δ_{i+1}:

$$velocity[\text{token}] = \{M \cdot velocity[\text{token}] - gradient\}, \quad \delta_{i+1}[\text{token}] = \{velocity[\text{token}]\} \tag{4.5}$$

Where M is a momentum factor used to control accumulated gradients. Finally, we apply δ_{i+1} on LLMs for next iteration. The revised generation process can be expressed as:

$$p_{LLM}(X_t \mid x_{<t}) = \text{softmax}[z_t - \delta_{i+1}] \tag{4.6}$$

The Kullback-Leibler (KL) Divergence is the objective function to compare word usage between the human-authored and LLM-revised corpora. This function quantifies overall

doi:10.6342/NTU202403360

Algorithm 1: Zeroth-order Bias Alignment Heuristics

Input: LLM , Human Corpus $C_h = \{c_i\}_{i=1}^{n}$, Instruction prompts x_{instruct}, Scaling factors: α, β, M
Output: Optimized δ for bias suppression, Revised Corpus C_l'

$min\_dl \leftarrow \infty$, $\delta_0 \leftarrow 0$, $velocity \leftarrow \varnothing$;
for $i \leftarrow 1$ **to** $max\_iterations$ **do**
  // Generate a new corpus with current δ_i
  $C_l' \leftarrow$ generate_corpus$(LLM, C_h, x_{\text{instruct}}, \delta_i)$;
  // Compute the objective function
  $L \leftarrow$ KL-Div(C_h, C_l') ;
  if $L < min\_dl$ **then**
    $min\_dl \leftarrow L$;
    if check_convergence(min_dl) **then**
      Break: ;
  // Update token frequency ratios
  $T_h \leftarrow \{\text{token} \mid \exists c \in C_h, \text{s.t. token} \in c\}$;
  $T'_l \leftarrow \{\text{token} \mid \exists c'_i \in C'_l, \text{s.t. token} \in c'_i\}$;
  $F'_l, F_H \leftarrow$ update_frequency_ratio(T'_l, T_h) ;
  // Gradient optimization for δ_{i+1}
  for token $\in$T'$_l \cup T_h$ **do**
    if token not in velocity **then**
      $velocity[\text{token}] \leftarrow 0$;
    $gradient \leftarrow \beta \cdot F'_l[\text{token}] - \alpha \cdot F_h[\text{token}]$;
    $velocity[\text{token}] \leftarrow M \cdot velocity[\text{token}] - gradient$;
    $\delta_{i+1}[\text{token}] \leftarrow velocity[\text{token}]$;

divergence and is defined as:

$$\mathcal{L} = KL(P_l \parallel P_h) = \textstyle\sum_{\text{token} \in W_h \cup W_l} P_l(\text{token}) \log \frac{P_l(\text{token})}{P_h(\text{token})} \tag{4.7}$$

Note that **words** are regarded as the basic elements in a paragraph, not tokens for echoing the issue that related works have raised. Therefore, $\mathcal{L}$ is selected as the objective function to minimize by controlling the model's token distribution with δ_{i+1}. Equivalently,

$$p_{LLM}(X_t \mid x_{<t}) \propto \frac{F_h}{F_l'} \propto \frac{P_h}{P_l'} \propto 1/L \tag{4.8}$$

These frequency ratios adjust token probabilities in the LLM by applying a `LogitsProcessor`, which controls the likelihood of each token without directly accessing the output logits. This method enhances human-preferred words and suppresses LLM-preferred words.

4.2 Discriminator Bootstrapped Nomination (DiBoN)

Here we present our second method: *Discriminator Bootstrapped Nomination (DiBoN)* which specializes in capturing semantic and syntactic bias of LLMs compared to *ZoBAH*,

doi:10.6342/NTU202403360

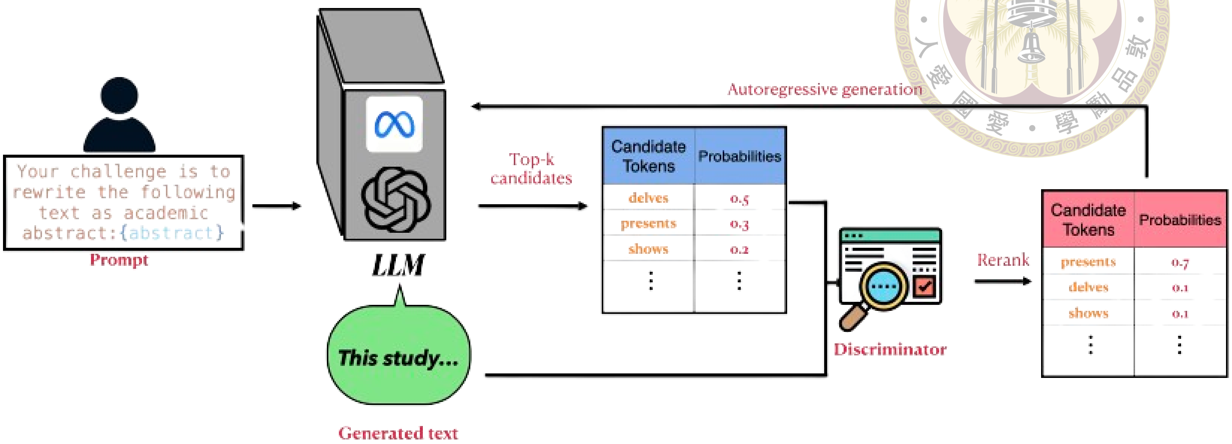


Figure 4.2: Overview of the Discriminator Bootstrapped Nomination (DiBoN)

whose overview is presented in Figure 4.2.

Inspired by guided-based methods like *PPLM* [16], we combine off-the-shelf AI-text detector [19], handcrafted syntactic[40] and surface features[13] to construct an ensemble discriminator D with the LLM and guide the LLM's generation process in inference time, as detailed in Algo 2.

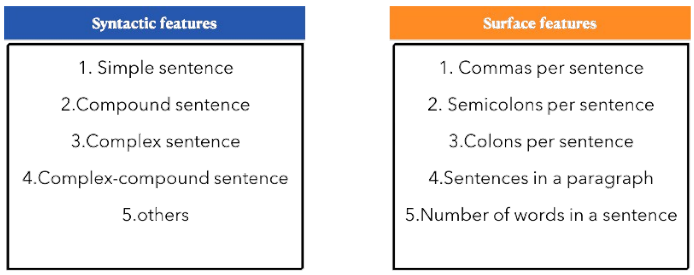


Figure 4.3: Handcrafted features for DiBoN

Detailed information about our handcrafted features is presented in Fig. 4.3, where numeric elements represent each feature. To aggregate this information, we concatenate the detector's last hidden state with current syntactic and surface features, employing a simple classification head to evaluate the text's scores. A higher score indicates a more remarkable similarity to our positive corpus.

At each generation step, we sample k token candidates and truncating undesirable ones from the LLM with their corresponding probabilities :

$$\{(x_t^j, p_t^j)\}_{j=1}^k = \text{Top-K}\left(\{(x_t, p_t) \mid x_t \in V, p_t = p_{LLM}(X_t \mid x_{<t})\}\right) \tag{4.9}$$

Next, we iterate these k token candidates and append the token with generated sequence $x_{<t}$, forming a new token sequence $\tilde{x}^j_{<t+1}$. By scanning each $\tilde{x}^j_{<t+1}$ with D, we can get the discriminator to score s_t^j, indicating the goodness of token x_t^j. Combining s_t^j with its

doi:10.6342/NTU202403360

original probability p_t^j, we can re-rank token candidates pool $\{(x_t^j, p_t^j)\}_{j=1}^k$, and fetch the token performing the best combination score $score_t^j$:

$$x_t = \arg\max_{x_t^j}(score_t^j) \tag{4.10}$$

Algorithm 2: Discriminator Bootstrapped Nomination

Input: LLM, Human Corpus $C_h = \{c_i\}_{i=1}^n$,Instruction prompts x_{instruct}, Discriminator D, Scaling factor λ

Output: Revised Corpus $C'_l = \{\tilde{c}_i\}_{i=1}^n$

for each document c_i in C_h **do**
    while $x_{t-1} = eod\_token\_id$ **do**
        $z_t \leftarrow$ generate_next_token$(LLM, x_{<t-1}, x_{\text{instruct}})$;
        $(x_t, p_t) = softmax[z_t]$
        `// Sample top-k token candidates`
        $\{(x_t^j, p_t^j)\}_{j=1}^k = \text{Top-K}\left(\{(x_t, p_t) = z_t \mid x_t \in V, p_t = p_{LLM}(X_t \mid x_{<t})\}\right)$;
        for each (x_t^j, p_t^j) in $\{(x_t^j, p_t^j)\}_{j=1}^k$ **do**
            $\tilde{x}_{<t+1}^j = \text{concat}(x_{<t}, x_t^j)$;
            $s_t^j = D(\tilde{x}_{<t+1}^j)$ `// Compute discriminator score`
            $\text{score}_t^j = p_t^j + \lambda s_t^j$ `// Combine scores`
        $x_t = \arg\max_{x_t^j}(\text{score}_t^j)$;
    $\tilde{c}_i \leftarrow x$;

doi:10.6342/NTU202403360

Chapter 5

Experiment

In this section, we outline the datasets, evaluation metrics, models and baselines used in our study.

5.1 Datasets

To rigorously evaluate the effectiveness of the **ZoBAH** and **DiBoN**, we utilize two benchmark datasets:

Field	**Number of Abstracts**
Computer Science	1815
EESS	49
Physics	49
Statistics	270
Math	215

Type	**Number of paragraphs**
L1 (Native English writer)	100
L2 (Non-native English writer)	100

Table 5.1: Summary of dataset abstracts collected from Multi-XScience (left) and the British Academic Written English Corpus (right)

- **Multi-XScience**[20]. This is is a large-scale multi-document dataset derived from scientific articles on arXiv, encompassing various domains. It includes abstracts and related work sections. For our study, abstracts were extracted from domains such as Computer Science, EESS, Physics, Statistics, and Math. Different document counts representing small, medium, and large corpora were used to validate the effectiveness of our methods across various scenarios.
- **British Academic Written English Corpus[21].** This is a collection of texts produced by undergraduate and Master's students across various disciplines for assess-

doi:10.6342/NTU202403360

ment in UK degree programs. To demonstrate our methods' ability to enhance the quality of L2 revisions, we extract paragraphs written by both native and non-native English writers.

5.2 Evaluation Metric

5.2.1 Lexical Metric

- **Term Frequency—Inverse Document Frequency (TF-IDF).** TF-IDF is used to compute document similarity by representing each document as a vector of TF-IDF scores, with each dimension representing a word's TF-IDF score. We then measure the similarity between the vectors of C_h and C_l'.
- **ROUGE[41].** This is a recall oriented metric counting the percentage of n-grams in the C_h that are present in C_l'. Here we use ROUGE-L, which refer to the matches of the longest common subsequence.
- **BLEU[42].** evaluates machine-translated text quality by comparing n-grams in the candidate text to those in reference translations. We consider the 1-4 gram overlap between C_h and C_l' to assess content similarity.

5.2.2 Pattern-based Metric

- **BLEURT[43].** BLEURT leverages pre-trained transformer-based models to evaluate the semantic similarity between the generated text and the reference. It goes beyond surface-level lexical matching to understand the deeper semantic and syntactic content.
- **Moverscorer[44].** Moverscorer leverages pre-trained word embeddings to measure the distance between words in a continuous vector space, allowing for a more nuanced evaluation than traditional lexical metrics.

5.2.3 AI-text Detection Rate

In addition to traditional metrics, we employ the state-of-the-art AI-text detector **Radar** [19] to evaluate the bias in generated results. This allows for a straightforward comparison between our methods and others, highlighting the effectiveness of our approach in reducing bias.

doi:10.6342/NTU202403360

5.3 Models

To simulate how researchers use LLMs for scientific writing revision, we selected two of the most common models: GPT-3.5 and LLAMA3-70B, representing a black-box LLM and a white-box LLM, respectively. For the detector within the discriminator in DiBoN, we use a smaller language model, Vicuna-7B, to guide LLM generation. This selection ensures that our methods are tested across various model architectures, enhancing their relevance and applicability to real-world scientific writing tasks.

5.4 Baseline

We evaluate our methods against the following three representative baselines:

- **Zero-shot prompting.** It is the most intuitive way for researchers to utilize LLMs for text revision. We simply use the prompt with the unpolished text:

  > *Your challenge is to rewrite the following academic abstract. Original abstract for reference:* ***unpolished text****. Please make the revised one distinguishable from the original. Revised abstract:*

- **WordBanning.** Like most prompt engineering works for text style transfer [45], we guide LLMs with additional instructions to control their text style. In our case, we mimic the instruction from [45] to do keyword exclusion to address the word-level bias identified in related research. We create a word list of terms excessively used by LLMs and instruct the LLMs to avoid these words:

  > *Your challenge is to rewrite the following academic abstract. Please mimic the style and nuances of human writing as closely as possible. Avoid using overly familiar phrases or typical AI-generated patterns. Use varied sentence structures, introduce subtle inconsistencies that are natural in human writing, and incorporate a unique voice. You are prohibited to use following words: Wordlist. Original abstract for reference:* ***unpolished text****. Please make the revised one distinguishable from the original. Revised abstract:*

- **Few-shot prompting.** Building on few-shot learning works for text style transfer [15, 31], we provide 4 examples—2 positive and 2 negative—for LLMs to learn the writing style of the positive set and avoid the negative set using their internal knowledge:

doi:10.6342/NTU202403360

*Your challenge is to rewrite the following academic abstract. Please mimic the style and nuances of human writing as closely as possible. Avoid using overly familiar phrases or typical AI-generated patterns. Use varied sentence structures, introduce subtle inconsistencies that are natural in human writing, and incorporate a unique voice. For instance, try to mimic the human-written style of following examples:**Positive examples**. And avoid the LLM-rewritten style with following example:**Negative examples** Original abstract for reference: **unpolished text**. Please make the revised one distinguishable from the original. Revised abstract:*

- **FUDGE[46].** Future Discriminators for Generation (FUDGE) is an auxiliary model that guides text generation toward or away from specific attributes by adjusting the logits for each token. For this approach, we utilize Vicuna-7B, which has been fine-tuned on 4,647 human-written abstracts sourced from NeurIPS, MIDL, and ICRL, to guide our base model.

- **DExperts[18].** Decoding-time Experts (DExperts) involve tuning a smaller language model (LM) and comparing the predictions of this tuned model (the expert) with its untuned counterpart (the anti-expert) to guide the larger base model. For the expert and anti-expert models, we use Vicuna-7B to guide our base model. The expert model is fine-tuned on 4,647 human-written abstracts, while the anti-expert model is fine-tuned on 4,647 LLM-generated abstracts collected from NeurIPS, MIDL, and ICRL.

- **PREADD[47].** Unlike other approaches that use auxiliary expert models to adjust for attributes, PREADD does not require an external model. Instead, it compares the output logits generated from a raw prompt with those from a prefix-added prompt, enabling positive and negative control over any attribute encapsulated by the prefix. In our setting, we use following prompt as prefix-added prompt to force model control the bias within text:

  Please ensure to write the text with your large language model style, incorporate the following preferred words:Wordlist

5.5 Experimental Setting

In our experiment, we utilize the zero-shot prompt outlined in 5.4 for both ZoBAH and DiBoN.

For ZoBAH, we configure the maximum iteration number to 30, with scaling factors α,

doi:10.6342/NTU202403360

β, and momentum M set to 0.5, 1.5, and 0.9, respectively. To ensure the optimization process converges, we terminate iterations when the absolute value of the loss function difference between consecutive iterations is less than 0.005.

Regarding DiBoN, we sample 10 candidates during the token selection process and activate our discriminator D when the length of $x_{<t}$ exceeds 100 to enhance computational efficiency. We set λ to 1.1818 to regulate the detector score. The generation process is halted when the last token in the sequence matches eod_token_id.

Finally, we fine-tune our discriminator D using 9,294 human-written abstracts collected from NeurIPS, MIDL, and ICRL conferences from 2020 to 2022, along with synthetic pairs. The fine-tuning process involves 3 training epochs with a learning rate of 1×10^{-5} and a per-device batch size of 8.

doi:10.6342/NTU202403360

Chapter 6

Results and Analysis

6.1 Evaluaion on the White-box LLM (LLAMA3-70B)

Method	**Lexical Metrics**			**Pattern-Based Metrics**		**Detection Rate**	**Inference**
	TF-IDF (↑)	**BLEU (↑)**	**ROUGE (↑)**	**BLEURT (↑)**	**MoverScore (↑)**	**Radar (↓)**	**Time (↓)***
Zero-shot	0.5562	0.1604	0.4113	0.3104	0.6170	0.6517	0.1667
WordBanning	0.5631	0.1736	0.4051	0.3234	0.6229	0.5951	0.1667
Few-shot	0.6070	0.2137	0.4511	0.3563	0.6396	0.6264	0.1667
FUDGE	0.7517	0.3669	0.5902	0.5317	0.6878	0.4445	11.40
DExperts	0.6512	0.2203	0.4728	0.4558	0.66981	0.66981	12.57
PREADD	0.6197	0.2366	0.4712	0.34927	0.6492	0.5613	22.13
CoDA	**0.7659**	**0.3769**	**0.6037**	**0.5321**	**0.6996**	**0.4239**	**8.891**
Human Corpus	-	-	-	-	-	0.2344	-

Table 6.1: Evaluation metrics for different methods across the Physics dataset. Metrics include lexical (TF-IDF, BLEU, ROUGE), pattern-based scorers (BLEURT, MoverScore), detection rate (Radar: LLM-detector), and inference time. *Inference Time indicates the time taken to generate a abstract on average(min/abstract).

Table 6.1 presents compelling evidence of CoDA's superiority over both prompt-based and guided-based approaches. Our method demonstrates remarkable improvements across lexical, pattern-based, and detection rate metrics compared to prompt engineering techniques. Notably, the CoDA reduces the detection rate by approximately 27% relative to Zero-shot prompting, underscoring its practical value in academic writing contexts. In comparison to other guided-based methods, CoDA excels in multiple aspects. It achieves the highest scores in lexical metrics (TF-IDF, BLEU, and ROUGE), indicating that its output more closely resembles human-written text. Furthermore, the CoDA significantly outperforms its competitors in evading detection, with the lowest Radar score of 0.4739. Crucially, CoDA accomplishes these improvements while maintaining computational efficiency. With an average inference time of 8.891 minutes per abstract, it operates 29%

doi:10.6342/NTU202403360

faster than its closest competitor, FUDGE (11.40 minutes), and significantly outpaces other methods like DExperts (12.57 minutes) and PREADD (22.13 minutes). These results collectively demonstrate CoDA's ability to generate high-quality, human-like text that effectively evades detection, all while maintaining a substantial speed advantage over existing guided-based approaches.

6.2 Evaluation on the Black-box LLM (GPT-3.5)

Method	Lexical Metrics			Pattern-Based Metrics		Detection Rate Metrics
	TF-IDF (↑)	BLEU (↑)	ROUGE (↑)	BLEURT (↑)	MoverScore (↑)	Radar (↓)
Zero-shot	0.7126	0.2400	0.5116	0.4634	0.6706	0.6754
WordBanning	0.6619	0.2468	0.4515	0.4238	0.6480	0.6779
Few-shot	0.6409	0.2225	0.4339	0.3959	0.6407	0.6803
Coda	**0.7179**	**0.2833**	**0.5249**	**0.4705**	**0.6721**	**0.6402**
Human Corpus	-	-	-	-	-	0.4341

Table 6.2: Evaluation metrics for different methods across the Statistc dataset using GPT-3.5. Metrics include lexical (TF-IDF, BLEU, ROUGE), pattern-based scorers (BLEURT, MoverScore), and detection rate (Radar: LLM-detector).

We conducted a comprehensive comparison of our methods against established baselines using the Statistics dataset in the black-box LLM scenario. Due to the inherent constraints of black-box LLMs, we applied only ZoBAH, limiting logits bias adjustments to 300. Despite these limitations, our approach achieved remarkable success in debiasing the LLM. The results clearly demonstrate that ZoBAH effectively reduces bias in generated outputs, underscoring its potential to enhance fairness and accuracy in language models even within restricted environments.

doi:10.6342/NTU202403360

6.3 Enhancing L2 Corpus Revision Quality

Method	Lexical Metric	Pattern-Based Metric	Detection Rate
	TF-IDF (↑)	Sentence-BERT (↑)	Radar (↓)
Original L2	0.02080	0.5669	**0.1522**
Zero-shot	0.01227	0.44476	0.6264
WordBanning	0.01429	0.5463	0.6231
Few-shot	0.01429	0.5463	0.5775
FUDGE	0.01721	0.5534	0.4071
DExperts	0.01667	0.5544	0.4227
PREADD	0.01843	0.56897	0.5613
Quillbot	0.02024	0.5641	0.5175
Grammarly	0.02048	0.5677	0.1727
CoDA	0.01677	0.5654	0.368535

Table 6.3: Evaluation of enhancement of CoDA with L2 dataset. Metrics include lexical (TF-IDF), pattern-based scorer (Sentence-BERT), and detection rate.

We evaluated CoDA's ability to enhance LLM-generated text using LLAMA3-70B to revise the L2 corpus. Table 6.3 compares CoDA with baselines and commercial tools, Quillbot and Grammarly[48, 49].

CoDA outperforms prompt-based and guided-based methods across all metrics, achieving higher TF-IDF (0.01893) and Sentence-BERT (0.5588) scores, indicating improved lexical and semantic similarity to the L1 corpus. Notably, CoDA's detection rate (Radar: 0.368535) is significantly lower than most baselines and Quillbot (0.5175), though slightly higher than Grammarly (0.1727). Compared to the original L2 corpus, CoDA closely approximates native-like quality while substantially reducing detectability. Its performance rivals commercial tools, demonstrating potential for practical applications in academic writing. While CoDA shows significant improvements, the slight differences from the original L2 corpus suggest room for further refinement in balancing quality enhancement with preservation of original text characteristics. These results highlight CoDA's effectiveness in improving LLM-generated text quality while maintaining low detectability, positioning it as a promising tool for academic and professional writing contexts.

doi:10.6342/NTU202403360

6.4 Case study: Generalizability of ZoBAH

Method	Lexical Metrics			Pattern-Based Metrics		Detection Rate
	TF-IDF (↑)	BLEU (↑)	ROUGE (↑)	BLEURT (↑)	MoverScore (↑)	Radar (↓)
ZoBAH (δ from EESS)	0.7119	0.2803	0.5176	0.4648	0.6710	0.6445
ZoBAH (δ from Physics)	0.7071	0.2792	0.5170	0.4648	0.6692	0.6492
ZoBAH (δ from Math)	0.7074	0.2781	0.5141	0.4560	0.6694	0.6430
ZoBAH (δ from CS)	**0.7181**	0.2788	0.5186	**0.4770**	0.6714	**0.6363**
ZoBAH(δ from Stat)	0.7179	**0.2833**	**0.5249**	0.4705	**0.6721**	0.6402
Human Corpus	-	-	-	-	-	0.4341

Table 6.4: Evaluation of generalizability of ZoBAH with Statistic dataset. Metrics include lexical (TF-IDF, BLEU, ROUGE), Pattern-based scorers (BLEURT, MoverScore), and detection rate (Radar: LLM-detector).

We explore the generalizability of ZoBAH in Table 6.4, where we apply the δ learned from different datasets and evaluate their transferability across domains. The results show that learning δ from a smaller dataset results in a slight drop in transferability, yet still offers significant improvement over our baseline. Interestingly, applying δ from a relatively larger dataset (in this case, CS) yields evaluation results comparable to or even surpassing those from the original domain. This indicates that with access to a sufficiently large and high-quality positive dataset, we can potentially construct a **universal** weight that effectively eliminates bias at both the word and pattern levels.

doi:10.6342/NTU202403360

Chapter 7

Conclusion

In this study, we have rigorously examined the phenomenon of familiar bias within large language models (LLMs) in the context of academic writing, mainly focusing on the prevalent usage of specific terms like "delve." Our findings illuminate the tendency of LLMs to produce text with reduced lexical and syntactic diversity, a significant concern for the academic community striving for nuanced and original scholarly communication.

Our proposed **CoDA** method presents a robust solution to this challenge. By leveraging positive and negative corpora to adjust logit biases in an offline training fashion, we have demonstrated the method's efficacy in mitigating word-level and pattern-level biases. This approach has shown promise in white-box LLMs and black-box models such as GPT-3.5-turbo, ensuring broader applicability and enhanced generalizability.

Through a series of meticulously designed experiments, we evaluated the **CoDA**'s performance across various datasets, including **Multi-XScience** and **BAWE**. The results affirm the method's capability to produce text that aligns more closely with human preferences regarding lexical diversity and semantic richness. The visualizations further underscore the domain shift achieved by our method, highlighting its potential to transform LLM outputs into more human-like, diverse, and contextually appropriate text.

Moreover, our study opens avenues for future research in several directions. We advocate further exploration of applying our methods to different types of models (like Gemini or Claude) and different tasks (text style transfer[50], detoxification[51] and personalization tasks[52]). Additionally, we will incorporate human evaluations to provide deeper insights into the qualitative improvements in writing style and coherence brought about by our method. Investigating the beam search with DiBoN and designing more fine-grained ZoBAH are also pivotal steps forward.

In conclusion, our research signifies a crucial step towards enhancing LLM-generated

doi:10.6342/NTU202403360

text's stylistic and lexical diversity, addressing the pressing issue of familiar bias. As LLMs continue to play an integral role in academic and professional writing, methods like **CoDA** will ensure that these models augment rather than stifle the creative and intellectual rigor inherent in human writing. Our work contributes to the academic discourse on LLMs. It paves the way for more sophisticated and human-like text generation, embodying the dynamic interplay between artificial intelligence and human creativity.

doi:10.6342/NTU202403360

References

[1] OpenAI. Openai models - gpt3.5, 2022.

[2] Xianjun Yang, Yan Li, Xinlu Zhang, Haifeng Chen, and Wei Cheng. Exploring the limits of chatgpt for query or aspect-based text summarization. *arXiv preprint arXiv:2302.08081*, 2023.

[3] keqin Peng, Liang Ding, Qihuang Zhong, Li Shen, Xuebo Liu, Min Zhang, Yuanxin Ouyang, and Dacheng Tao. Towards making the most of chatGPT for machine translation. In *The 2023 Conference on Empirical Methods in Natural Language Processing*, 2023.

[4] Suzanne Fergus, Michelle Botha, and Mehrnoosh Ostovar. Evaluating academic answers generated using chatgpt. *Journal of Chemical Education*, 100(4):1672–1675, 2023.

[5] John Flowerdew. Some thoughts on english for research publication purposes (erpp) and related issues. *Language Teaching*, 48(2):250–262, 2015.

[6] Sung Il Hwang, Joon Seo Lim, Ro Woon Lee, Yusuke Matsui, Toshihiro Iguchi, Takao Hiraki, and Hyungwoo Ahn. Is chatgpt a "fire of prometheus" for non-native english-speaking researchers in academic writing? *Korean Journal of Radiology*, 24(10):952, 2023.

[7] Elinor Poole-Dayan, Deb Roy, and Jad Kabbara. Llm targeted underperformance disproportionately impacts vulnerable users. *arXiv preprint arXiv:2406.17737*, 2024.

[8] Weixin Liang, Mert Yuksekgonul, Yining Mao, Eric Wu, and James Zou. Gpt detectors are biased against non-native english writers. *Patterns*, 4(7), 2023.

[9] Weixin Liang, Zachary Izzo, Yaohui Zhang, Haley Lepp, Hancheng Cao, Xuandong Zhao, Lingjiao Chen, Haotian Ye, Sheng Liu, Zhi Huang, Daniel McFarland, and James Y. Zou. Monitoring AI-modified content at scale: A case study on the impact

doi:10.6342/NTU202403360

of chatGPT on AI conference peer reviews. In *Forty-first International Conference on Machine Learning*, 2024.

[10] Weixin Liang, Yaohui Zhang, Zhengxuan Wu, Haley Lepp, Wenlong Ji, Xuandong Zhao, Hancheng Cao, Sheng Liu, Siyu He, Zhi Huang, et al. Mapping the increasing use of llms in scientific papers. *arXiv preprint arXiv:2404.01268*, 2024.

[11] Dmitry Kobak, Rita González Márquez, Emoke-Agnes Horvát, and Jan Lause. Delving into chatgpt usage in academic writing through excess vocabulary. *arXiv e-prints*, pages arXiv–2406, 2024.

[12] Xiaofei Wang, Hayley M Sanders, Yuchen Liu, Kennarey Seang, Bach Xuan Tran, Atanas G Atanasov, Yue Qiu, Shenglan Tang, Josip Car, Ya Xing Wang, et al. Chatgpt: promise and challenges for deployment in low-and middle-income countries. *The Lancet Regional Health–Western Pacific*, 41, 2023.

[13] Bakhtiyar Syed, Gaurav Verma, Balaji Vasan Srinivasan, Anandhavelu Natarajan, and Vasudeva Varma. Adapting language models for non-parallel author-stylized rewriting. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 9008–9015, 2020.

[14] Akhila Yerukola, Xuhui Zhou, Elizabeth Clark, and Maarten Sap. Don't take this out of context! on the need for contextual models and evaluations for stylistic rewriting. *arXiv preprint arXiv:2305.14755*, 2023.

[15] Avanti Bhandarkar, Ronald Wilson, Anushka Swarup, and Damon Woodard. Emulating author style: A feasibility study of prompt-enabled text stylization with off-the-shelf llms. In *Proceedings of the 1st Workshop on Personalization of Generative AI Systems (PERSONALIZE 2024)*, pages 76–82, 2024.

[16] Sumanth Dathathri, Andrea Madotto, Janice Lan, Jane Hung, Eric Frank, Piero Molino, Jason Yosinski, and Rosanne Liu. Plug and play language models: A simple approach to controlled text generation. *arXiv preprint arXiv:1912.02164*, 2019.

[17] Ben Krause, Akhilesh Deepak Gotmare, Bryan McCann, Nitish Shirish Keskar, Shafiq Joty, Richard Socher, and Nazneen Fatema Rajani. Gedi: Generative discriminator guided sequence generation. *arXiv preprint arXiv:2009.06367*, 2020.

[18] Alisa Liu, Maarten Sap, Ximing Lu, Swabha Swayamdipta, Chandra Bhagavatula, Noah A Smith, and Yejin Choi. Dexperts: Decoding-time controlled text generation with experts and anti-experts. *arXiv preprint arXiv:2105.03023*, 2021.

doi:10.6342/NTU202403360

[19] Xiaomeng Hu, Pin-Yu Chen, and Tsung-Yi Ho. Radar: Robust ai-text detection via adversarial learning. *Advances in Neural Information Processing Systems*, 36:15077–15095, 2023.

[20] Yao Lu, Yue Dong, and Laurent Charlin. Multi-xscience: A large-scale dataset for extreme multi-document summarization of scientific articles. *arXiv preprint arXiv:2010.14235*, 2020.

[21] Hilary Nesi, Sheena Gardner, Paul Thompson, Paul Wickens, et al. British academic written english corpus. *Oxford Text Archive Core Collection*, 2008.

[22] Hanqing Zhang, Haolin Song, Shaoyu Li, Ming Zhou, and Dawei Song. A survey of controllable text generation using transformer-based pre-trained language models. *ACM Computing Surveys*, 56(3):1–37, 2023.

[23] Nitish Shirish Keskar, Bryan McCann, Lav R Varshney, Caiming Xiong, and Richard Socher. Ctrl: A conditional transformer language model for controllable generation. *arXiv preprint arXiv:1909.05858*, 2019.

[24] Alvin Chan, Yew-Soon Ong, Bill Pung, Aston Zhang, and Jie Fu. Cocon: A self-supervised approach for controlled text generation. *arXiv preprint arXiv:2006.03535*, 2020.

[25] Karin de Langis, Ryan Koo, and Dongyeop Kang. Reinforcement learning with dynamic multi-reward weighting for multi-style controllable generation. *arXiv preprint arXiv:2402.14146*, 2024.

[26] Di Jin, Zhijing Jin, Zhiting Hu, Olga Vechtomova, and Rada Mihalcea. Deep learning for text style transfer: A survey. *Computational Linguistics*, 48(1):155–205, 2022.

[27] Tianxiao Shen, Tao Lei, Regina Barzilay, and Tommi Jaakkola. Style transfer from non-parallel text by cross-alignment. *Advances in neural information processing systems*, 30, 2017.

[28] Minh Tran, Yipeng Zhang, and Mohammad Soleymani. Towards a friendly online community: An unsupervised style transfer framework for profanity redaction. *arXiv preprint arXiv:2011.00403*, 2020.

[29] Wei Xu, Alan Ritter, William B Dolan, Ralph Grishman, and Colin Cherry. Paraphrasing for style. In *Proceedings of COLING 2012*, pages 2899–2914, 2012.

[30] Harsh Jhamtani, Varun Gangal, Eduard Hovy, and Eric Nyberg. Shakespearizing modern language using copy-enriched sequence-to-sequence models. *arXiv preprint arXiv:1707.01161*, 2017.

doi:10.6342/NTU202403360

[31] Emily Reif, Daphne Ippolito, Ann Yuan, Andy Coenen, Chris Callison-Burch, and Jason Wei. A recipe for arbitrary text style transfer with large language models. *arXiv preprint arXiv:2109.03910*, 2021.

[32] Tyler A Chang and Benjamin K Bergen. Language model behavior: A comprehensive survey. *Computational Linguistics*, 50(1):293–350, 2024.

[33] Eric Michael Smith, Melissa Hall, Melanie Kambadur, Eleonora Presani, and Adina Williams. " i'm sorry to hear that": Finding new biases in language models with a holistic descriptor dataset. *arXiv preprint arXiv:2205.09209*, 2022.

[34] Stephanie Brandl, Ruixiang Cui, and Anders Søgaard. How conservative are language models? adapting to the introduction of gender-neutral pronouns. *arXiv preprint arXiv:2204.10281*, 2022.

[35] Kaitlyn Zhou, Kawin Ethayarajh, and Dan Jurafsky. Richer countries and richer representations. *arXiv preprint arXiv:2205.05093*, 2022.

[36] Nikita Nangia, Clara Vania, Rasika Bhalerao, and Samuel R Bowman. Crows-pairs: A challenge dataset for measuring social biases in masked language models. *arXiv preprint arXiv:2010.00133*, 2020.

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

[38] Christopher Manning and Hinrich Schutze. *Foundations of statistical natural language processing*. MIT press, 1999.

[39] Yoshua Bengio, Réjean Ducharme, and Pascal Vincent. A neural probabilistic language model. *Advances in neural information processing systems*, 13, 2000.

[40] Song Feng, Ritwik Banerjee, and Yejin Choi. Characterizing stylistic elements in syntactic structure. In *Proceedings of the 2012 joint conference on empirical methods in natural language processing and computational natural language learning*, pages 1522–1533, 2012.

[41] Chin-Yew Lin. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81, 2004.

[42] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 311–318, 2002.

doi:10.6342/NTU202403360

[43] Thibault Sellam, Dipanjan Das, and Ankur P Parikh. Bleurt: Learning robust metrics for text generation. *arXiv preprint arXiv:2004.04696*, 2020.

[44] Wei Zhao, Maxime Peyrard, Fei Liu, Yang Gao, Christian M Meyer, and Steffen Eger. Moverscore: Text generation evaluating with contextualized embeddings and earth mover distance. *arXiv preprint arXiv:1909.02622*, 2019.

[45] Yihan Chen, Benfeng Xu, Quan Wang, Yi Liu, and Zhendong Mao. Benchmarking large language models on controllable generation under diversified instructions. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 17808–17816, 2024.

[46] Kevin Yang and Dan Klein. Fudge: Controlled text generation with future discriminators. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3511–3535, 2021.

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

[48] Marzuki, Utami Widiati, Diyenti Rusdin, Darwin, and Inda Indrawati. The impact of ai writing tools on the content and organization of students' writing: Efl teachers' perspective. *Cogent Education*, 10(2):2236469, 2023.

[49] Mike Perkins. Academic integrity considerations of ai large language models in the post-pandemic era: Chatgpt and beyond. *Journal of University Teaching and Learning Practice*, 20(2), 2023.

[50] Sudha Rao and Joel Tetreault. Dear sir or madam, may i introduce the gyafc dataset: Corpus, benchmarks and metrics for formality style transfer. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pages 129–140, 2018.

[51] Samuel Gehman, Suchin Gururangan, Maarten Sap, Yejin Choi, and Noah A Smith. Realtoxicityprompts: Evaluating neural toxic degeneration in language models. *arXiv preprint arXiv:2009.11462*, 2020.

[52] Alireza Salemi, Sheshera Mysore, Michael Bendersky, and Hamed Zamani. Lamp: When large language models meet personalization. *arXiv preprint arXiv:2304.11406*, 2023.

doi:10.6342/NTU202403360

Appendix A — Ablation study of CoDA

Method	Lexical Metrics			Pattern-Based Metrics		Detection Rate
	TF-IDF (↑)	**BLEU (↑)**	**ROUGE (↑)**	**BLEURT (↑)**	**MoverScore (↑)**	**Radar (↓)**
ZoBAH	0.6474	0.2560	0.4861	0.3919	0.6452	0.5991
DiBoN	0.6227	0.2342	0.4483	0.4149	0.6528	0.6290
CoDA	**0.6687**	**0.3060**	**0.5229**	**0.4855**	**0.6601**	**0.5798**
Human Corpus	-	-	-	-	-	0.3560

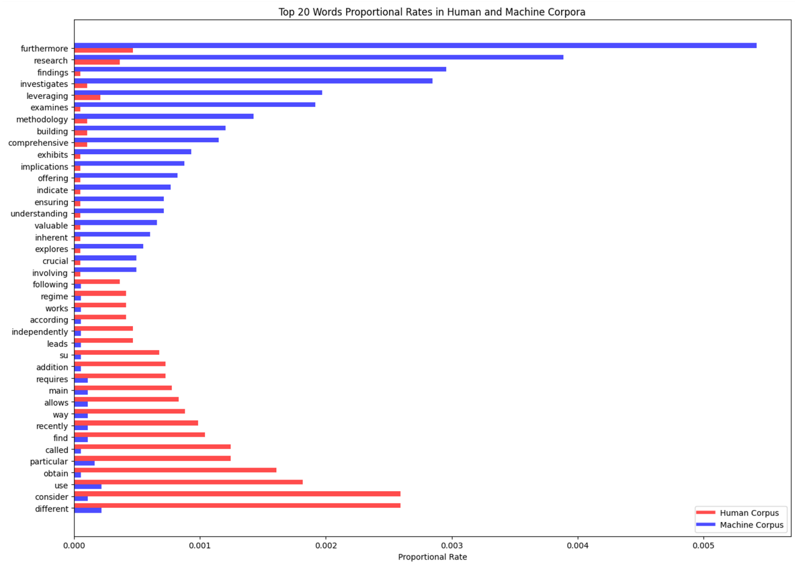
Table A.1: Ablation study for our methods with Llama3-70B across the EESS dataset. Metrics include lexical (TF-IDF, BLEU, ROUGE), model-based scorers (BLEURT, MoverScore), and detection rate (Radar: LLM-detector).

This section presents an ablation study examining the roles of two key components, ZoBAH and DiBoN, within CoDA. Table A.1 shows that ZoBAH performs better in lexical metrics, while DiBoN excels in pattern-based metrics, aligning with our design principles. Interestingly, in the detection rate analysis, ZoBAH outperforms DiBoN, suggesting that lexical elements are more significant than pattern-based features for AI-text detectors.

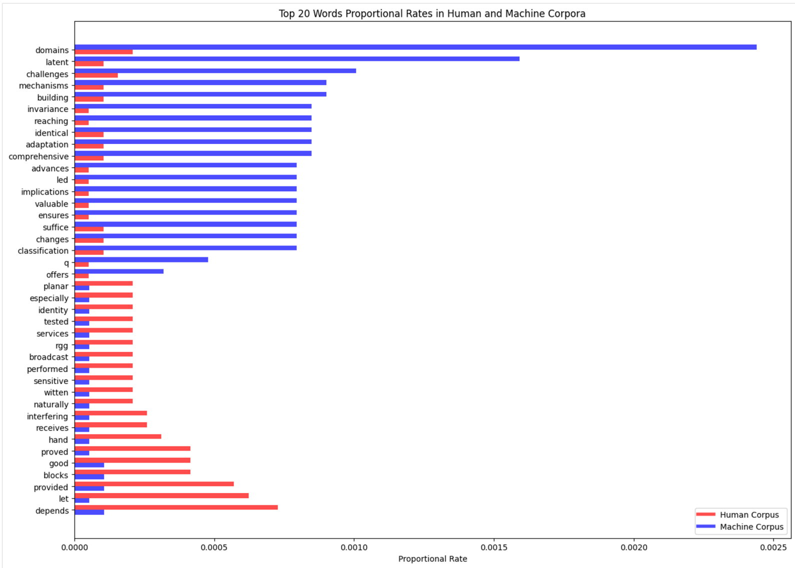
doi:10.6342/NTU202403360

doi:10.6342/NTU202403360

Appendix B — Visualization of Bias Mitigation



(a) Top 20 biased words in zero-shot corpus



(b) Top 20 biased words in CoDA corpus

Figure B.1: Word usage for zero-shot and CoDA corpus

doi:10.6342/NTU202403360

As illustrated in Figure B.1, we visualize the top 20 LLAMA3-biased words in the revised Math corpus for both zero-shot and CoDA approaches. Our method effectively reduces excessive word usage by nearly 50%. This significant reduction not only validates our objective of alleviating word-level bias but also underscores the efficacy of CoDA in addressing issues identified in related studies. These results highlight our method's potential to enhance the fairness and accuracy of LLM-generated text, reinforcing its applicability in mitigating biases in academic writing.

doi:10.6342/NTU202403360