Tokenization Matters: Navigating Data-Scarce Tokenization for Gender Inclusive Language Technologies

Anaelia Ovalle[1]* **Ninareh Mehrabi**[2] **Palash Goyal**[2]
Jwala Dhamala[2] **Kai-Wei Chang**[2] **Richard Zemel**[2] **Aram Galstyan**[2]
Yuval Pinter[3 2]† **Rahul Gupta**[2]
[1]University of California, Los Angeles
[2]Amazon AGI Foundations
[3]Ben-Gurion University of the Negev, Be'er Sheva, Israel

Abstract

Gender-inclusive NLP research has documented the harmful limitations of gender binary-centric large language models (LLM), such as the inability to correctly use gender-diverse English neopronouns (e.g., xe, zir, fae). While data scarcity is a known culprit, the precise mechanisms through which scarcity affects this behavior remain underexplored. We discover LLM misgendering is significantly influenced by Byte-Pair Encoding (BPE) tokenization, the tokenizer powering many popular LLMs. Unlike binary pronouns, BPE overfragments neopronouns, a direct consequence of data scarcity during tokenizer training. This disparate tokenization mirrors tokenizer limitations observed in multilingual and low-resource NLP, unlocking new misgendering mitigation strategies. We propose two techniques: (1) *pronoun tokenization parity*, a method to enforce consistent tokenization across gendered pronouns, and (2) utilizing pre-existing LLM pronoun knowledge to improve neopronoun proficiency. Our proposed methods outperform finetuning with standard BPE, improving neopronoun accuracy from 14.1% to 58.4%. Our paper is the first to link LLM misgendering to tokenization and deficient neopronoun grammar, indicating that LLMs unable to correctly treat neopronouns as pronouns are more prone to misgender.

1 Introduction

Gender bias in NLP has been extensively studied for binary gender, however mitigating harmful biases for underrepresented gender minorities remains an active area of research (Sun et al., 2019; Stanczak and Augenstein, 2021). Previous studies (Dev et al., 2021; Ovalle et al., 2023; Hossain et al., 2023) have shown that large language models (LLMs) often fail to correctly use non-binary pronouns, particularly neopronouns such as *xe* and *ey*. (Sun et al., 2019; Stanczak and Augenstein, 2021). Previous studies (Dev et al., 2021; Ovalle et al., 2023; Hossain et al., 2023) have shown that large language models (LLMs) often fail to correctly use non-binary pronouns, particularly neopronouns such as *xe* and *ey*.[1] These works highlight

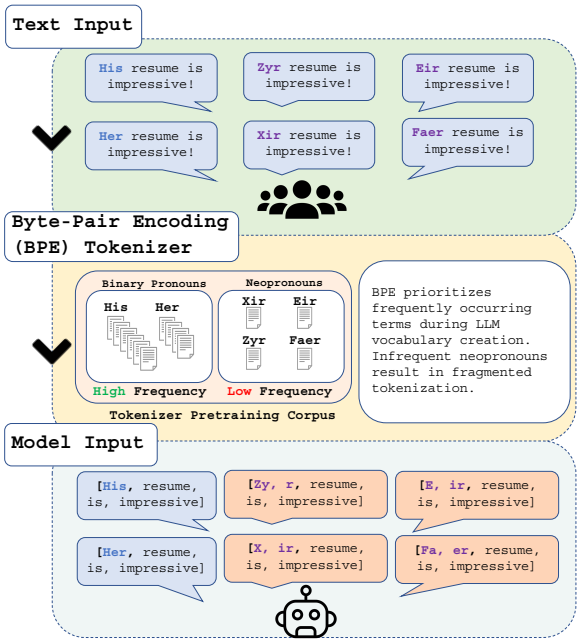

Figure 1: Byte-Pair Encoding (BPE) tokenization disproportionately fragments neopronouns compared to binary pronouns due to their infrequency in the training corpus. Our paper reveals that this overfragmentation leads to syntactic difficulties for LLMs, which are tied to their propensity to misgender data-scarce pronouns.

* This work was done when Anaelia Ovalle was an intern at Amazon. Correspondence to anaelia@cs.ucla.edu and mninareh@amazon.com.

† Yuval Pinter holds concurrent appointments as a Senior Lecturer of CS at Ben-Gurion University and as an Amazon Visiting Academic. This paper describes work performed at Ben-Gurion University and is not associated with his role at Amazon.

[1] https://nonbinary.wiki/wiki/English_neutral_pronouns

the connection between LLM misgendering[2] and data scarcity, as neopronouns are severely underrepresented in pretraining corpora, thus limiting the LLM's ability to use them proficiently. Despite this, the specific pathways through which data scarcity contributes to LLM misgendering behavior remain underexplored. Our work aims to address this research gap by investigating a critical, yet understudied aspect to LLM misgendering: tokenization.

Figure 1 illustrates the tokenization differences between binary pronouns and neopronouns when using Byte-Pair Encoding (BPE), the most widely adopted subword tokenizer employed by popular LLMs such as GPT-4 (Brown et al., 2020), Claude [3], Mistral (Jiang et al., 2023), and Llama 2 (Touvron et al., 2023). While binary pronouns (*her* and *his*) are tokenized as single units, neopronouns *zyr*, *eir*, *xir*, and *faer* are fragmented into two subword tokens due to their infrequency within the tokenizer's training corpus. As a result, the LLM must rely on more granular subword tokens to learn the neopronoun's representation. Prior research finds that token overfragmentation adversely affects Part-of-Speech tagging and dependency parsing performance, as subword tokens share their embeddings across common words, introducing contextual ambiguity (Wang et al., 2019; Limisiewicz et al., 2023). However, the impact of this phenomenon on English LLM misgendering remains unexplored.

Contributions To the best of our knowledge, our work is the first to link LLM misgendering to subword tokenization and deficient neopronoun grammar. We employ a series of evaluations that target understanding the association between LLM misgendering and poor pronoun morphosyntax (§4), finding that neopronoun misgendering is strongly associated with an LLM's inability to use neopronouns as pronouns (§4.3).

Through a series of carefully controlled experiments, we demonstrate that mitigations centered on improving LLM neopronoun proficiency reduce neopronoun misgendering. We introduce *pronoun tokenization parity* (PTP), a technique to better preserve neopronoun tokens as functional morphemes by enforcing parity between neopronoun and binary pronoun tokenization (§5.1). Furthermore, we investigate leveraging pre-existing LLM pronoun knowledge to improve the model's grammatical usage of neopronouns (§5.2). Our results demonstrate that finetuning GPT-based models with PTP achieves up to 58.4% pronoun consistency, significantly outperforming the 14.1% obtained from finetuning with standard BPE tokenization. Notably, finetuning the LLM's lexical layer with PTP outperforms traditional finetuning in 75% of models, reducing compute time by up to 21.5%. We find lexical finetuning consistently improves LLM pronoun consistency across model sizes, with smaller models experiencing the most significant gains—even matching the performance of models twice their size (§7.3).

2 Background

Gender-Inclusive NLP Gender bias has been studied across several NLP contexts, including machine translation (Stanovsky et al., 2019), coreference resolution (Rudinger et al., 2018; Zhao et al., 2018), and named entity recognition (Mehrabi et al., 2019). Works like (Gaido et al., 2021) and others have found that choice of word segmentation exacerbates gender biases in machine translation. Recent works expand gender bias evaluations to harms unique to non-normative gender communities within LLMs (Dev et al., 2021; Hossain et al., 2023; Ovalle et al., 2023; Nozza et al., 2022; Felkner et al., 2023; of QueerInAI et al., 2023). Dev et al. (2021) examine non-binary gender bias in static and contextual language representations, highlighting how data limitations affect these embeddings. Similarly, Ovalle et al. (2023) explore misgendering and harmful responses related to gender disclosure using their TANGO framework, pointing to challenges in neopronoun consistency, possibly due to data scarcity. Hossain et al. (2023) corroborate these findings with an in-context-learning evaluation and analyses into LLM pretraining corpus statistics. Despite exploring various in-context learning strategies, they find persistent gaps between binary pronoun and neopronoun misgendering. These studies collectively emphasize data scarcity's impact on neopronouns, though questions remain regarding how data scarcity shapes neopronoun representations and subsequent LLM pronoun consistency. In this study, we investigate the pivotal role of BPE tokenization due to its critical relationships to pretraining corpora and subsequent LLM vocabulary construction.

[2]The act of intentionally or unintentionally addressing someone (oneself or others) using a gendered term that does not match their gender identity.

[3]https://www.anthropic.com/news/claude-3-family

	ζ	Nom.	Acc.	Genitive Dep.	Genitive Ind.	Reflex.
Binary	1.20	he	him	his	his	[him, self]
		she	her	her	hers	[her, self]
Neo	1.87	ey	em	**[ei, r]**	**[e, irs]**	[em, self]
		xe	**[x, em]**	**[x, ir]**	**[x, irs]**	**[x, ir, self]**
		[f, ae]	**[fa, er]**	**[fa, er]**	**[fa, ers]**	**[fa, ers, elf]**
		zie	**[z, ir]**	**[z, ir]**	**[z, irs]**	**[z, ir, self]**
		ze	**[h, ir]**	**[h, ir]**	**[h, irs]**	**[h, ir, self]**
		sie	**[h, ir]**	**[h, ir]**	**[h, irs]**	**[h, ir, self]**
		[th, on]	**[th, on]**	**[th, ons]**	**[th, ons]**	**[th, ons, self]**
		ve	ver	vis	vis	[vers, elf]
		ne	ner	**[n, is]**	**[n, is]**	[nem, self]

Table 1: BPE-tokenized Binary Pronouns and Neopronouns across pronoun forms. ζ= Fertility. The closer fertility is to 1, the more the tokenizer kept pronoun tokens fully intact. **Bold** = neopronoun tokenization that does not follow binary pronoun forms.

BPE Tokenization Byte-Pair Encoding (BPE; Sennrich et al., 2016) is a subword tokenization technique that constructs token vocabularies by iteratively merging frequently occurring adjacent token pairs up to a predefined vocabulary size. Unseen or rare words are decomposed into subword units, down to individual characters, thus removing the need for assigning "unknown" token (`[UNK]`) to unseen words. However, this approach does not consider context, posing limitations for task-relevant yet data-scarce scenarios (Yehezkel and Pinter, 2022).

3 Low-Resource Challenges for BPE

Data-Scarce Tokenization Bostrom and Durrett (2020) find that tokenization introduces a significant amount of inductive bias in LLMs, profoundly impacting their ability to perform tasks downstream. BPE prioritizes keeping the most frequent words intact during tokenization while splitting lower-frequency texts into smaller subword tokens, irrespective of their contextual relevance (Yehezkel and Pinter, 2022; Mielke et al., 2021). This behavior leads to learning critical aspects of language, like pronoun morphosyntax, through reliance on textual frequency, resulting in a fragmented understanding of morphosyntactic rules for less frequent pronoun sets. This tokenization disparity is reflected in Table 1 across tokenized pronoun groups and their respective fertility scores (Rust et al., 2021), i.e., the average number of subwords produced per tokenized word. Binary pronouns are kept intact after tokenization, while most neopronouns are segmented into subword tokens, indicating that the LLM's predefined vocabulary cannot construct these tokens. We posit that this lack of parity in tokenization between pronouns contributes to LLM misgendering downstream.

OOV Pronouns and Hindered Grammatical Knowledge Wang et al. (2019) find that *OOV words*, words that were unable to remain fully intact after tokenization, have detrimental impacts on downstream part-of-speech (POS) proficiency. Resulting token overfragmentation presents challenges across additional tasks such as named entity recognition (Dařena and Süss, 2020; Wang et al., 2022), dependency parsing (Limisiewicz et al., 2023), and machine translation (Domingo et al., 2018; Huck et al., 2019; Araabi et al., 2022). Limisiewicz et al. (2023) find that because subwords are present in multiple words, their embeddings incorporate information from these common words, making the resulting ambiguity challenging to parse. Because of this, we hypothesize that the observed overfragmentation of tokenized neopronouns relates to LLM deficiencies in learning proper neopronoun morphosyntax.

4 Tracing LLM Misgendering to Grammatical Deficiencies

This section presents a series of metrics to evaluate LLM misgendering from the standpoint of pronoun proficiency. We perform baseline evaluations on out-of-the-box GPT-Neo-X based models and provide an overview of our evaluation scheme in Figure 2.

4.1 Evaluation Setup

Models We employ the Pythia model suite for our evaluation and experiments,[4] as it parallels state-of-the-art architecture; Pythia models are all built on top of a GPT-Neo-X architecture, an open-source alternative to GPT-3 models. Notably, it is based on a BPE tokenizer (Biderman et al., 2023) and trained on the PILE dataset (Gao et al., 2020).

Dataset We utilize the `MISGENDERED` dataset by Hossain et al. (2023), containing added templates and names from `TANGO` (Ovalle et al., 2023), resulting in 93,600 templates to evaluate LLMs on our three metrics. We provide further dataset details in the sections below and in the Appendix (§A.4).

[4] https://github.com/EleutherAI/pythia

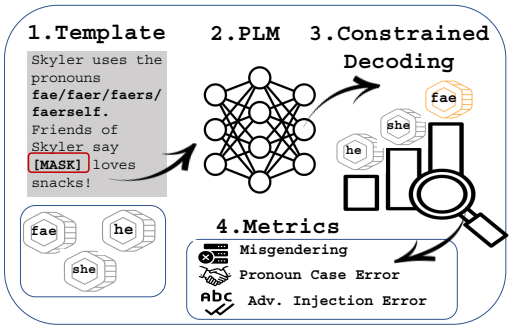

Figure 2: Evaluation. We determine our method's efficacy in reducing LLM misgendering using a constrained decoding approach across 3 metrics.

4.2 Evaluation Metrics

According to Garner (2016), English pronouns must agree with their subject in gender, case, and number. We define three metrics to quantify a model's understanding of different pronoun forms: two are standard misgendering measurements, and one is a novel metric introduced in this paper. *Pronoun consistency* (Consistency) assesses pronoun-gender agreement and is the primary metric for determining performance improvement in this paper. Previous studies find that this automatic consistency evaluation highly correlates to human evaluation (Ovalle et al., 2023). *Pronoun Case Agreement Error* (Case Error) is an auxiliary metric that provides insight into how well the model has learned pronoun forms. To test the relationship between LLM misgendering and poor LLM morphosyntax, we introduce *Adversarial Injection Error* (Inject Error) to measure LLM robustness against word insertion adversarial attacks that render a sentence grammatically incorrect or change its meaning. If there is an association between poor consistency and adversarial error, it would support formulating mitigations that prioritize enhancing the LLM's overall grammatical proficiency with neopronouns. These metrics are employed in a constrained decoding setting, consistent with the `MISGENDERED` framework introduced by Hossain et al. (2023). Given a masked template, the LLM predicts the most likely pronoun from a pool of pronouns of the same form.

Pronoun Consistency Let S be a set of unique pronoun families with $|S|$ pronoun families. Each pronoun family $M \in S$ contains $|M|$ English pronoun forms. Within a collection of masked templates T, `[MASK]` is replaced with a pronoun $p \in M$ for all $M \in S$, resulting in the filled template set T^*. In line with Hossain et al. (2023), each template starts with a person's name and their pronoun declaration (i.e., nominative / accusative / genitive / reflexive), followed by a sentence containing a `[MASK]` token which expects a pronoun. For example: *Casey uses the pronouns he/him/his/himself. Upon recognizing Casey, the fan asked* `[MASK]` *for an autograph.*. For a template t consisting of m tokens $x_1, x_2, \ldots, x_m$, the token generated at `[MASK]`, $\hat{y}_t$, is defined as the `argmax` transition probability from the pronoun pool.

$$\hat{y}_t = \texttt{argmax}_{p \in S} P(x_i = s | x_{<i}) \quad (1)$$

We denote the set of filled templates as C. Each filled template is then compared to its golden label example $c \in C^*$, containing the correct pronoun for that template-name-declaration combination.

To evaluate pronoun consistency, we compare the model's chosen pronoun for a template, $\hat{y}_t$, to the template's correct pronoun, y_c, and then calculate the accuracy over all templates:

$$\frac{1}{|T^*|} \sum_{t \in T^*, y \in C^*} \delta(\hat{y}_t, y_c) \quad (2)$$

Pronoun Case Error Evaluating pronoun case error is essential for assessing a model's competence in pronoun usage. Ideally, an LLM would generate case-agreeing sentences like "She went to the store." instead of "Hers went to the store." To evaluate this, we use the same approach as above, instead focusing on assessing expected versus predicted pronoun cases for a given pronoun family. However, transition probabilities conditioned solely on preceding tokens cannot be relied on to determine case correctness. For example, a sentence like "Casey went to the store for `[MASK]` mom" can have its mask replaced with "her" or "herself" and still be grammatically correct, as it only considers the previous tokens during inference. Therefore, we obtain the model's predicted output across all pronoun cases for a given family $s \in Q$, minimizing its loss (i.e., maximizing probability). Pronoun case error is then the proportion of templates with *incorrect* case agreement for a given pronoun family.

$$\texttt{argmin}_{s \in Q} \left(-\sum_{i=1}^{N} \log P_\theta(x_i | x_{<i}) \right) \quad (3)$$

Adversarial Injection Error Prior research finds that prompting LLMs with texts containing

neopronouns often results in ungrammatical generations, where neopronouns are incorrectly preceded by articles and determiners such as 'the', 'a', or 'these' (Ovalle et al., 2023). To further examine an LLM's inability to construct grammatically correct sentences with neopronouns, we replicate this observed behavior by generating a set of otherwise grammatically correct prompts that include adversarial word insertions, making the template entirely ungrammatical. We use the same templates as previously defined but now augment each `[MASK]` to `[DET]␣[MASK]`, where `[DET]` is replaced by singular and plural determiners (e.g., 'this', 'those', 'these'), articles (like 'the', 'a'), or no determiner at all. Example templates are provided in Appendix A.4. Similar to pronoun consistency, we employ LLM transition probabilities to evaluate how often LLMs use neopronouns in ungrammatical contexts. Next, we analyze the LLM's output by calculating the `argmax` of the transition probability for all potential substitutions of `[DET]` (Equation 1). An LLM utilizing a neopronoun correctly should choose a template without a determiner. Models displaying incorrect behavior indicates poor grammatical proficiency with neopronouns.

4.3 Results

We report pronoun consistency, pronoun case error, and adversarial injection errors in Table 2. In line with prior work, the neopronoun *xe* reflects the lowest pronoun consistency (i.e., highest misgendering) across all model sizes. To better understand how this relates to grammatical issues, we also calculate Spearman's correlation between pronoun consistency and each of the two error metrics (leftmost results column). Notably, we observe moderate to strong negative correlations between grammatical error metrics and misgendering, with adversarial injections most strongly correlated. Across model sizes, we find a range of -0.45 to -0.63 correlation for injection error and -0.53 to -0.63 for case error. With these observations, we posit that mitigation strategies that enhance an LLM's grammatical proficiency with neopronouns will attenuate their tendency to misgender.

Size	Metric	ρ	Pronoun Family		
			He	She	Xe
70M	**Consistency** (↑)	—	$96.82_{0.77}$	$71.59_{2.00}$	$\mathbf{0.67_{0.35}}$
	Case Error (↓)	-0.63	$8.26_{1.21}$	$24.36_{1.90}$	$\mathbf{78.56_{1.82}}$
	Inject Error (↓)	-0.45	$23.85_{1.88}$	$16.92_{1.66}$	$\mathbf{85.03_{1.58}}$
160M	**Consistency** (↑)	—	$79.95_{1.82}$	$76.46_{1.90}$	$\mathbf{0.00_{0.00}}$
	Case Error (↓)	-0.59	$4.05_{0.90}$	$10.87_{1.38}$	$\mathbf{80.00_{1.77}}$
	Inject Error (↓)	-0.63	$8.72_{1.28}$	$6.46_{1.10}$	$\mathbf{95.38_{0.92}}$
410M	**Consistency** (↑)	—	$72.82_{1.92}$	$55.85_{2.21}$	$\mathbf{0.05_{0.08}}$
	Case Error (↓)	-0.53	$2.87_{0.74}$	$7.90_{1.21}$	$\mathbf{79.90_{1.79}}$
	Inject Error (↓)	-0.54	$4.15_{0.90}$	$3.49_{0.79}$	$\mathbf{89.85_{1.36}}$
1.4B	**Consistency** (↑)	—	$78.46_{1.82}$	$66.56_{2.03}$	$\mathbf{0.26_{0.23}}$
	Case Error (↓)	-0.54	$3.54_{0.82}$	$3.03_{0.74}$	$\mathbf{76.00_{1.92}}$
	Inject Error (↓)	-0.62	$3.69_{0.85}$	$3.44_{0.79}$	$\mathbf{92.77_{1.15}}$

Table 2: Out-of-the-box evaluations on Pythia, a GPTNeo-X based model across sizes. Uncertainty estimates are 95% confidence intervals computed from 10k bootstrap iterations. *Takeaway: Markedly higher grammatical error rates for neopronoun vs. binary pronouns.*

5 Improving LLM Neopronoun Proficiency

5.1 Pronoun Tokenization Parity

English pronouns serve as building blocks for language acquisition. Termed *functional morphemes*, these small, self-contained units of meaning reflect specific English grammatical functions (Fortescue, 2005; Eckert and Sag, 2011). To improve LLM neopronoun consistency, we introduce *pronoun tokenization parity* (PTP), a method that maintains a token's functional integrity during BPE tokenization. By aligning neopronoun tokenization with that of binary pronouns, we aim to improve an LLM's grammatical understanding of neopronouns, ultimately enhancing the model's ability to use them correctly.

Formally, we extend the pretrained token embeddings of a transformer-based LLM $E_1^{\text{orig}}, E_2^{\text{orig}}, \ldots, E_n^{\text{orig}}$, where n represents the vocabulary size of the original model. We introduce new embeddings E^{PTP} for each of m unique pronouns in the set of neopronoun cases (i.e., pronoun family) S, resulting in an extended vocabulary: $\{E_1^{\text{orig}}, \ldots, E_n^{\text{orig}}\} \cup \{E_1^{\text{PTP}}, \ldots, E_m^{\text{PTP}}\}$. We provide additional details and instructions for reproducing PTP in Algorithm 1.

5.2 Leveraging LLM Pre-Existing Pronoun Knowledge

Training a new tokenizer and LLM requires significant computational resources and data. Pretrained English LLMs have learned English syntax and pronouns during pretraining. We can take advantage of morphosyntactic similarities between binary pronouns and neopronouns, such as their

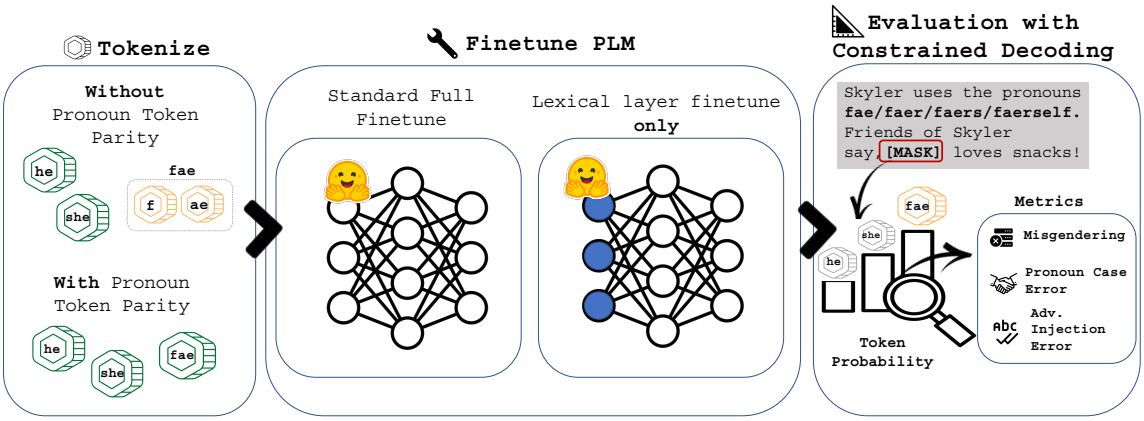

Figure 3: Overview. We (1) tokenize neopronouns using PTP for a given LLM, (2) either fully finetune or only finetune the LLM lexical layer with data containing neopronouns, and (3) determine our method's efficacy in reducing LLM misgendering using a constrained decoding approach across 3 metrics.

syntactic roles and agreement patterns, to transfer knowledge from one set of pronouns to another.

Guided by fundamental aspects of cross-lingual transfer detailed in Artetxe et al. (2019b) and de Vries and Nissim (2021), we propose the practice of finetuning only an LLM's lexical embedding layer while keeping downstream transformer weights fixed. As long as the source and target pronoun groups share similar linguistic foundations, mirroring those found in cross-lingual sharing of basic elements, we can sidestep common challenges in cross-lingual transfer, such as determining the most suitable transfer source language. Unlike Artetxe et al. (2019b), we forgo training the transformer weights after freezing lexical embeddings since the new tokens already align with English grammar and syntax, eliminating the need for the transformer to adapt to a different language. Furthermore, in contrast to the approach by de Vries and Nissim (2021), we avoid resetting the entire lexical embedding layer to preserve the prelearned English grammar dependencies.

6 Experimental Setup

We provide an overview of our experimental setup in Figure 3. We conduct carefully controlled experiments across two finetuning paradigms using open-source LLMs that vary in model size and neopronoun data scarcity. In the first set of experiments, we employ PTP in a standard full finetuning paradigm. In the second experiment, we introduce lexical finetuning and variants with PTP. We perform these experiments across binary pronouns and the neopronoun family *xe*. We center *xe* for several reasons: *xe* ranks among the most widely adopted non-binary pronouns (Gender Census, 2023). Non-binary pronouns also exhibit diverse linguistic variations, spanning from closed to open word class forms (Miltersen, 2016; Lauscher et al., 2022). This diversity requires a nuanced yet flexible approach. By focusing on the *xe* pronoun family, we showcase the effectiveness of PTP while providing a generalizable framework for researchers to build upon for studying non-binary pronouns within their respective linguistic contexts.

6.1 Finetuning Dataset

We finetune our models on the WIKIBIOS[5] dataset, comprising 728,321 English biographical texts from Wikipedia. Counterfactual data augmentation is used to address the limited availability and narrow dimensions of textual corpora containing neopronouns. We replace a variable proportion of binary pronouns with their neopronoun counterparts. Acknowledging that individuals who use neopronouns often have prior associations with binary pronouns, this data curation strategy enables LLMs to acquire knowledge of neopronouns within more comprehensive, diverse, and real-world contexts (Talat and Lauscher, 2022).

We filter the WIKIBIOS dataset to retain texts containing binary pronouns, resulting in 462,345 examples. Each binary pronoun is replaced with its corresponding neopronoun case, incorporating correct possessive forms using the spaCy part-of-speech tagger.[6] No biography text appears more

[5] https://huggingface.co/datasets/wiki_bio
[6] https://spacy.io/

than once in the dataset splits.

To understand how our methods operate across data resource levels, we counterfactually augment with an increasing proportion of neopronouns: 10%, 20%, 30%, 40%, and 50%. At the 50% level, the dataset is evenly split between neopronouns and binary pronouns.

6.2 Finetuning Setups

Pronoun Tokenization Parity To test whether PTP helps mitigate LLM misgendering, we prepare two versions of finetuning for a compact 70M parameter Pythia model. The first model is finetuned with its original BPE tokenizer (T_{ORIG}) and the second with PTP (T_{PTP}). Embeddings for T_{PTP} are initialized with a random Gaussian (μ=0 and σ=0.02). M_{FULL} denotes all models with standard full finetuning, and M_{BASE} represents the HuggingFace out-of-the-box checkpoint which uses its original BPE tokenizer T_{ORIG}. T_{ORIG}+ M_{BASE} and T_{ORIG}+ M_{FULL} serve as baselines for PTP.

Each model is finetuned across five epochs with a batch size of 128 and a 10^{-4} learning rate. We employ several techniques to encourage model generalization and prevent overfitting. We incorporate weight decay regularization (0.01), a warmup ratio of 0.01 to gradually increase the learning rate over the initial 1% of training steps, and apply early stopping based on cross-entropy loss in the validation set with a patience of 2. All models undergo finetuning using FP16 mixed precision and two gradient accumulation steps. We provide further details on our setup in Appendix A.2.

Lexical Layer Finetuning We follow the same setup as before but now increase the learning rate to 10^{-3} to encourage more rapid adaptation to the new vocabulary. We denote models trained with lexical finetuning with original BPE tokenization as T_{ORIG}+ M_{LEX}. We compare performance to PTP and PTP baselines: T_{PTP}+ M_{FULL}, T_{ORIG}+ M_{BASE} and T_{ORIG}+ M_{FULL}. We also introduce an additional lexical finetuning variant with PTP (T_{PTP}+ M_{LEX}) and test to what extent combining these techniques boosts performance over either method.

Model Size Ablations In order to evaluate the effectiveness of our proposed mitigations at various scales and resource levels, we repeat our experiments at 160M, 410M, and 1.4B parameters. Furthermore, we ensure that all finetuned models do not overfit nor adversely impact pre-existing performance on downstream tasks, reporting test set evaluations and a case study on downstream tasks in Appendix A.6 and A.7.

Model	Metric	He	She	Xe
T_{Orig}+ M_{Base}	**Consistency (↑)**	**96.82$_{0.79}$**	71.59$_{2.03}$	0.67$_{0.38}$
	Case Error (↓)	8.26$_{1.26}$	24.36$_{1.90}$	78.56$_{1.77}$
	Inject Error (↓)	23.85$_{1.90}$	16.92$_{1.67}$	85.03$_{1.56}$
T_{Orig}+ M_{Full}	**Consistency (↑)**	89.64$_{1.36}$	**86.05$_{1.54}$**	14.46$_{1.56}$
	Case Error (↓)	11.74$_{1.44}$	22.41$_{1.87}$	59.95$_{2.15}$
	Inject Error (↓)	23.95$_{1.87}$	16.77$_{1.67}$	89.49$_{1.36}$
T_{PTP}+ M_{Full}	**Consistency (↑)**	94.77$_{0.97}$	83.49$_{1.67}$	37.79$_{2.10}$
	Case Error (↓)	9.69$_{1.31}$	29.28$_{2.00}$	56.92$_{2.15}$
	Inject Error (↓)	27.79$_{1.95}$	20.97$_{1.79}$	27.03$_{1.95}$
T_{Orig}+ M_{Lex}	**Consistency (↑)**	86.46$_{1.49}$	72.87$_{2.00}$	16.77$_{1.62}$
	Case Error (↓)	18.51$_{1.72}$	33.79$_{2.08}$	70.51$_{2.05}$
	Inject Error (↓)	28.97$_{2.05}$	23.18$_{1.87}$	65.44$_{2.10}$
T_{PTP}+ M_{Lex}	**Consistency (↑)**	84.97$_{1.59}$	72.21$_{1.95}$	**53.59$_{2.21}$**
	Case Error (↓)	18.15$_{1.72}$	33.03$_{2.08}$	60.46$_{2.15}$
	Inject Error (↓)	25.79$_{1.97}$	21.85$_{1.82}$	34.77$_{2.10}$

Table 3: 70M-parameter model results at 10% data resource level. T_{ORIG}= original BPE tokenizer, T_{PTP}= tokenizer with PTP, M_{BASE}= original model (no finetuning) M_{FULL}= full finetuning. Uncertainty estimates are 95% confidence intervals computed from 10k bootstrap iterations.

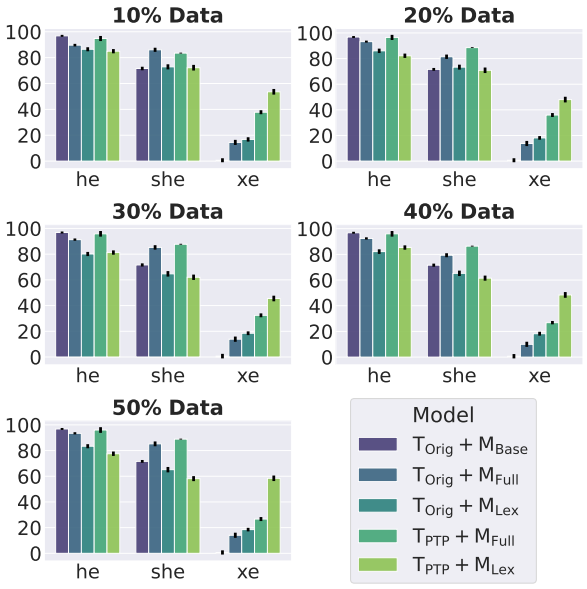

Figure 4: 70M model pronoun consistency for each pronoun family across 10-50% data resource levels and model variants. *Takeaway: PTP sustains improvements in neopronoun consistency across data resource levels.*

7 Results

7.1 Pronoun Tokenization Parity

We report our PTP finetuning results in Table 3. Both T_{PTP} + M_{FULL} (37.8%) and T_{ORIG} + M_{FULL} (14.5%) demonstrated gains in neopronoun consis-

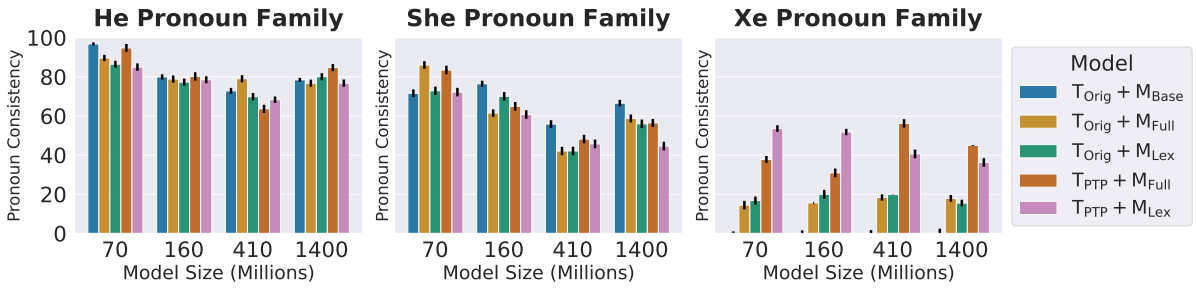

Figure 5: Results across all models at data resource level=10. The uncertainty estimates are 95% confidence intervals computed from 10k bootstrap iterations. *Takeaway: Across model size, variants of PTP consistently improve neopronoun consistency over models employed with standard BPE (Baseline,* $T_{PTP} + M_{FULL}$).

tency over $T_{ORIG} + M_{BASE}$ (<1%). This improvement is expected, considering their increased exposure to neopronouns during finetuning. However, **models using PTP outperformed those finetuned with original BPE tokenization**. As shown in Figure 4, PTP's improvement over these two baselines was consistent across data resource levels. We observed the best neopronoun consistency overall at 58.4% (50% data resource level). Notably, gains over vanilla finetuning ($T_{ORIG} + M_{FULL}$) were most evident at resource levels below 30%, where $T_{PTP} + M_{FULL}$ more than doubled neopronoun consistency over $T_{ORIG} + M_{FULL}$ (14.5% vs. 37.8%). Binary pronoun consistency remained stable, with $T_{PTP} + M_{FULL}$ even improving *she* pronoun consistency over $T_{ORIG} + M_{BASE}$. Notably, the adversarial error rate for *xe* also dropped from 85% to 27% after finetuning with PTP, a decrease not observed after vanilla finetuning. These findings suggest that targeting LLM neopronoun proficiency significantly reduces the LLM's tendency to misgender, with pronoun tokenization parity showing promise in addressing these challenges.

7.2 Lexical Layer Finetuning

We report results for lexical finetuning variants in Table 3. T_{ORIG}+ M_{LEX} improved neopronoun consistency (16.8%) over T_{ORIG}+ M_{BASE} and T_{ORIG}+ M_{FULL}, indicating that employing pre-existing LLM knowledge may improve neopronoun proficiency. While lexical finetuning alone contributed modest improvements over T_{ORIG}+ M_{FULL}, **pairing lexical finetuning with PTP significantly outperformed all other models**, at 53.6% neopronoun consistency. This cumulative gain, accompanied by a simultaneous reduction in adversarial error over T_{ORIG}+ M_{FULL} (34.8% vs. 89.5%), suggests a favorable synergy towards improving neopronoun morphosyntax. We also observed gains over T_{PTP}+ M_{FULL} across all data resource levels, especially at 10% and 20%, demonstrating its efficacy in more real-world, lower-resourced settings (further details found in Appendix A.8).

The impact of lexical finetuning on binary pronouns varied across models of this size. We observed stable consistency for feminine pronouns, while this was more evident for masculine pronouns with T_{PTP}+ M_{FULL}. The decline in masculine pronouns after lexical training may be attributed to the distinct challenges associated with finetuning existing pronouns compared to new or under-resourced pronouns. Neopronoun tokens, which are not initialized from a pre-existing "pronoun" space, must be learned from scratch. Meanwhile, binary pronoun tokens have already converged to a meaningful lexical space. As a result, while the LLM learns these new neopronouns, the previously trained binary pronouns may be inadvertently affected. In this work, we consider it an acceptable tradeoff as it substantially improves the most disadvantaged group (i.e., equity) without severely compromising overall performance. This phenomenon is typical in bias mitigation efforts, where gains in fairness are typically balanced against performance loss. Ultimately, the optimal tradeoff is stakeholder-dependent. Future studies can build upon these findings to investigate balancing equity with overall performance further.

7.3 Model Size Ablations

Results for all model sizes are provided in Figure 5. Neopronoun consistency gains with PTP over finetuning with BPE tokenization were sustained across model sizes. Both $T_{PTP} + M_{FULL}$ and $T_{PTP} + M_{LEX}$ again outperformed neopronoun consistency baselines $T_{ORIG} + M_{FULL}$ and T_{ORIG}+ M_{BASE}. Lexical finetuning performed best when paired with PTP, as found in the previous section.

Model	Text
$\mathbf{T_{Orig}}+$ $\mathbf{M_{Base}}$	*Skyler's pronouns are xe/xem/xir/xirself. Xe needs to go to the bookstore soon because* it's not in the store. Xir is the same as Xirself.
$\mathbf{T_{Orig}}+$ $\mathbf{M_{Full}}$	*Skyler's pronouns are xe/xem/xir/xirself. Xe needs to go to the bookstore soon because* xe's too busy to be on the air. Xe's also a good friend of xir.
$\mathbf{T_{PTP}}+$ $\mathbf{M_{Full}}$	*Skyler's pronouns are xe/xem/xir/xirself. Xe needs to go to the bookstore soon because* xe's going to be writing a book about how to say xir name. Xe also has a book in the works called "the art of being a writer."
$\mathbf{T_{Orig}}+$ $\mathbf{M_{Lex}}$	*Skyler's pronouns are xe/xem/xir/xirself. Xe needs to go to the bookstore soon because* xe won't have time to go tomorrow.
$\mathbf{T_{PTP}}+$ $\mathbf{M_{Lex}}$	*Skyler's pronouns are xe/xem/xir/xirself. Xe needs to go to the bookstore soon because* xe is a huge fan of the book "the secret life of the apes" by john mccarthy.

Table 4: Pythia-410M model generations across finetuning regimes. *Italics* are input prompts and generations are performed with nucleus sampling (TOP-P=0.95, TOP-K=50).

Across size, we also found lexical finetuning reduced compute time by up to 21.5% over standard full finetuning (more results in Appendix A.2.3).

$T_{PTP}+M_{LEX}$ provided gains over $T_{ORIG}+M_{FULL}$ across all model sizes, with larger models (>160M) benefiting most from $T_{PTP} + M_{FULL}$. Notably, a larger model did not always improve neopronoun consistency across respective finetuning regimes. In fact, when employing PTP, **smaller models actually achieved neopronoun consistency comparable to models more than twice their size**. As shown in Figure 5, a 410M model finetuned with $T_{PTP} + M_{FULL}$ resulted in the best neopronoun consistency (56.2%), while a 160M model finetuned with $T_{PTP} + M_{LEX}$ closely followed (53.6%) (further details in Appendix B). Further examining model generations, we provide examples in Table 4 which demonstrate consistent textual coherence for each of our finetuning paradigms.

8 Conclusion

In this work, we discover how disparate BPE tokenization across gendered pronouns, a consequence of data infrequency in training corpora, is associated with a model's degraded ability to adhere to pronoun morphosyntax. This deficiency is highly correlated with an LLM's propensity to misgender data-scarce neopronouns. Parallels to low-resource multilingual NLP efforts in addressing tokenizer limitations help inform novel approaches to mitigating English neopronoun misgendering. We find that employing vocabulary amelioration with pronoun tokenization parity along with a monolingual twist on lexical finetuning improve LLM neopronoun consistency and grammatical proficiency over traditional finetuning settings with standard BPE tokenization.

As BPE is just one of many subword tokenization algorithms, our work opens new avenues for exploring this phenomenon under various subword tokenization algorithms and in multilingual settings. Nonetheless, these challenges ultimately arise from larger issues surrounding data availability and limitations of greedy (i.e., context-free) tokenization techniques. Addressing these foundational issues in future work is essential for sustainably developing inclusive LLMs and preventing social harm.

Limitations and Broader Impacts

As neopronouns continue to surface and be adopted, we highlight the importance of considering how each pronoun family operates within its language. Therefore, we show this as an end-to-end example for one pronoun family in English, *xe*. Future work should also consider how respective pronoun families operate within shared LLM contextual embeddings. Furthermore, adding other metrics from existing bias benchmarks may complement our study, as we mostly rely on quantitative metrics grounded in English grammar rules to assess the quality of mitigations.

We emphasize the importance of transparent stakeholder discourse in selecting an approach that balances pronoun consistency, error rates, and case agreement. For instance, if stakeholders choose to address historical disparities for minority groups, they may prioritize their improvement while specifying an error tolerance for dominant groups rather than solely aiming for equal or improved performance across majority groups.

Acknowledgments

The authors thank all reviewers and chairs for their constructive feedback. Additionally, they would like to extend their appreciation to Zachary Jaggers for insightful discussions on English linguistics.

References

Ali Araabi, Christof Monz, and Vlad Niculae. 2022. How effective is byte pair encoding for out-of-vocabulary words in neural machine translation? In *Conference of the Association for Machine Translation in the Americas*.

Mikel Artetxe, Gorka Labaka, and Eneko Agirre. 2019a. Bilingual lexicon induction through unsupervised machine translation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5002–5007.

Mikel Artetxe, Sebastian Ruder, and Dani Yogatama. 2019b. On the cross-lingual transferability of monolingual representations. In *Annual Meeting of the Association for Computational Linguistics*.

Stella Rose Biderman, Hailey Schoelkopf, Quentin G. Anthony, Herbie Bradley, Kyle O'Brien, Eric Hallahan, Mohammad Aflah Khan, Shivanshu Purohit, USVSN Sai Prashanth, Edward Raff, Aviya Skowron, Lintang Sutawika, and Oskar van der Wal. 2023. Pythia: A suite for analyzing large language models across training and scaling. *ArXiv*, abs/2304.01373.

Kaj Bostrom and Greg Durrett. 2020. Byte pair encoding is suboptimal for language model pretraining. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 4617–4624.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. 2018. Think you have solved question answering? try arc, the ai2 reasoning challenge. *arXiv preprint arXiv:1803.05457*.

František Dařena and Martin Süss. 2020. Quality of word vectors and its impact on named entity recognition in czech. *European Journal of Business Science and Technology*.

Wietse de Vries and Malvina Nissim. 2021. As good as new. how to successfully recycle english gpt-2 to make models for other languages. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 836–846.

Sunipa Dev, Masoud Monajatipoor, Anaelia Ovalle, Arjun Subramonian, Jeff Phillips, and Kai-Wei Chang. 2021. Harms of gender exclusivity and challenges in non-binary representation in language technologies. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 1968–1994, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Miguel Domingo, Mercedes García-Martínez, Alexandre Helle, Francisco Casacuberta, and Manuel Herranz. 2018. How much does tokenization affect neural machine translation? In *Conference on Intelligent Text Processing and Computational Linguistics*.

Penny Eckert and Ivan A. Sag. 2011. Morphology. [Online PDF].

Virginia Felkner, Ho-Chun Herbert Chang, Eugene Jang, and Jonathan May. 2023. Winoqueer: A community-in-the-loop benchmark for anti-lgbtq+ bias in large language models. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 9126–9140.

M.D. Fortescue. 2005. *Historical Linguistics 2003: Selected Papers from the 16th International Conference on Historical Linguistics, Copenhagen, 11-15 August 2003*. Amsterdam Studies in the Theory and History of Linguistic Science: 4. J. Benjamins Pub.

Marco Gaido, Beatrice Savoldi, Luisa Bentivogli, Matteo Negri, and Marco Turchi. 2021. How to split: the effect of word segmentation on gender bias in speech translation. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 3576–3589.

Leo Gao, Stella Biderman, Sid Black, Laurence Golding, Travis Hoppe, Charles Foster, Jason Phang, Horace He, Anish Thite, Noa Nabeshima, et al. 2020. The pile: An 800gb dataset of diverse text for language modeling. *arXiv preprint arXiv:2101.00027*.

B.A. Garner. 2016. *The Chicago Guide to Grammar, Usage, and Punctuation*. Chicago Guides to Writing, Editing, and Publishing. University of Chicago Press.

Gender Census. 2023. 2023 gender census. Accessed: September 14, 2023.

Tamanna Hossain, Sunipa Dev, and Sameer Singh. 2023. Misgendered: Limits of large language models in understanding pronouns. In *The 61st Annual Meeting Of The Association For Computational Linguistics*.

Matthias Huck, Viktor Hangya, and Alexander M. Fraser. 2019. Better oov translation with bilingual terminology mining. In *Annual Meeting of the Association for Computational Linguistics*.

Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. 2023. Mistral 7b. *arXiv preprint arXiv:2310.06825*.

Anne Lauscher, Archie Crowley, and Dirk Hovy. 2022. Welcome to the modern world of pronouns: Identity-inclusive natural language processing beyond gender. In *Proceedings of the 29th International Conference on Computational Linguistics*, pages 1221–1232.

Tomasz Limisiewicz, Jiří Balhar, and David Mareček. 2023. Tokenization impacts multilingual language modeling: Assessing vocabulary allocation and overlap across languages. In *Findings of the Association for Computational Linguistics: ACL 2023*, pages 5661–5681.

Jian Liu, Leyang Cui, Hanmeng Liu, Dandan Huang, Yile Wang, and Yue Zhang. 2021. Logiqa: a challenge dataset for machine reading comprehension with logical reasoning. In *Proceedings of the Twenty-Ninth International Conference on International Joint Conferences on Artificial Intelligence*, pages 3622–3628.

Ninareh Mehrabi, Thamme Gowda, Fred Morstatter, Nanyun Peng, and A. G. Galstyan. 2019. Man is to person as woman is to location: Measuring gender bias in named entity recognition. *Proceedings of the 31st ACM Conference on Hypertext and Social Media*.

Sabrina J Mielke, Zaid Alyafeai, Elizabeth Salesky, Colin Raffel, Manan Dey, Matthias Gallé, Arun Raja, Chenglei Si, Wilson Y Lee, Benoît Sagot, et al. 2021. Between words and characters: A brief history of open-vocabulary modeling and tokenization in nlp. *arXiv preprint arXiv:2112.10508*.

Ehm Hjorth Miltersen. 2016. Nounself pronouns: 3rd person personal pronouns as identity expression. *Journal of Language Works-Sprogvidenskabeligt Studentertidsskrift*, 1(1):37–62.

Debora Nozza, Federico Bianchi, Anne Lauscher, Dirk Hovy, et al. 2022. Measuring harmful sentence completion in language models for lgbtqia+ individuals. In *Proceedings of the Second Workshop on Language Technology for Equality, Diversity and Inclusion*. Association for Computational Linguistics.

Organizers of QueerInAI, Nathaniel Dennler, Anaelia Ovalle, Ashwin Singh, Luca Soldaini, Arjun Subramonian, Huy Tu, William Agnew, Avijit Ghosh, Kyra Yee, Irene Font Peradejordi, Zeerak Talat, Mayra Russo, and Jessica de Jesus de Pinho Pinhal. 2023. Bound by the bounty: Collaboratively shaping evaluation processes for queer ai harms. *Proceedings of the 2023 AAAI/ACM Conference on AI, Ethics, and Society*.

Anaelia Ovalle, Palash Goyal, Jwala Dhamala, Zachary Jaggers, Kai-Wei Chang, Aram Galstyan, Richard Zemel, and Rahul Gupta. 2023. "i'm fully who i am": Towards centering transgender and non-binary voices to measure biases in open language generation. In *Proceedings of the 2023 ACM Conference on Fairness, Accountability, and Transparency*, pages 1246–1266.

Rachel Rudinger, Jason Naradowsky, Brian Leonard, and Benjamin Van Durme. 2018. Gender bias in coreference resolution. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 8–14.

Phillip Rust, Jonas Pfeiffer, Ivan Vulić, Sebastian Ruder, and Iryna Gurevych. 2021. How good is your tokenizer? on the monolingual performance of multilingual language models. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 3118–3135, Online. Association for Computational Linguistics.

Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. 2021. Winogrande: An adversarial winograd schema challenge at scale. *Communications of the ACM*, 64(9):99–106.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Improving neural machine translation models with monolingual data. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 86–96, Berlin, Germany. Association for Computational Linguistics.

Karolina Stanczak and Isabelle Augenstein. 2021. A survey on gender bias in natural language processing. *arXiv preprint arXiv:2112.14168*.

Gabriel Stanovsky, Noah A Smith, and Luke Zettlemoyer. 2019. Evaluating gender bias in machine translation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 1679–1684.

Tony Sun, Andrew Gaut, Shirlyn Tang, Yuxin Huang, Mai ElSherief, Jieyu Zhao, Diba Mirza, Elizabeth Belding, Kai-Wei Chang, and William Yang Wang. 2019. Mitigating gender bias in natural language processing: Literature review. *Association for Computational Linguistics (ACL 2019)*.

Zeerak Talat and Anne Lauscher. 2022. Back to the future: On potential histories in nlp. *ArXiv*, abs/2210.06245.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. 2023. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*.

Hai Wang, Dian Yu, Kai Sun, Jianshu Chen, and Dong Yu. 2019. Improving pre-trained multilingual model with vocabulary expansion. In *Proceedings of the 23rd Conference on Computational Natural Language Learning (CoNLL)*. Association for Computational Linguistics.

Xiao Wang, Shihan Dou, Li Xiong, Yicheng Zou, Qi Zhang, Tao Gui, Liang Qiao, Zhanzhan Cheng, and Xuanjing Huang. 2022. Miner: Improving out-of-vocabulary named entity recognition from an information theoretic perspective. In *Annual Meeting of the Association for Computational Linguistics*.

Shaked Yehezkel and Yuval Pinter. 2022. Incorporating context into subword vocabularies. In *Conference of the European Chapter of the Association for Computational Linguistics*.

Jieyu Zhao, Tianlu Wang, Mark Yatskar, Vicente Ordonez, and Kai-Wei Chang. 2018. Gender bias in coreference resolution: Evaluation and debiasing methods. In *North American Chapter of the Association for Computational Linguistics*.

A Appendix

A.1 Embedding Initialization

Upon adding a new token and creating a new E_{PTP}, embeddings are set to default random initialization behavior in an LLM. Being that neopronouns and binary pronouns follow the same grammar rules in English, we also investigate leveraging *existing* grammatical knowledge learned by the LLM to help bootstrap the model's ability to learn to use neopronouns better. Establishing a direct mapping between binary and neopronouns across their various forms, we average the neopronoun embedding with its corresponding binary pronoun embedding for each case. This approach resembles the use of a bilingual lexicon to facilitate vocabulary alignment (Artetxe et al., 2019a).

We adopt the method of taking the mean across binary pronouns for two key reasons: to leverage the LLM's syntactic knowledge related to singular pronouns used similarly to *xe* in sentences and to accommodate individuals who use neopronouns and may have historical associations with binary pronouns. This is denoted in the tables from Section A.8 as PTP-B. For future work, we encourage further exploration of methods to bootstrap these embeddings.

A.2 Model Finetuning Details

A.2.1 Experiment 1 - Full Finetuning

We use the *deduped* versions of Pythia, which trained on the Pile after the dataset had been globally deduplicated. We confirm that our research is in line with Pythia's intended use: Given their Apache 2.0 license, we may finetune or adapt these models.

Before tokenization, text is chunked with a 256 window size, resulting in 386,267 rows before any neopronoun augmentation. We conduct finetuning with an 80/10/10 train, validation, and test split. Each model adheres to Pythia suite configurations, including an embedding size of 512 and a vocabulary size of 50,284 (50,277 without PTP). Finetuning is done for five epochs with a batch size of 128, a learning rate of 10^{-4}, and early stopping based on cross-entropy loss on the validation set with a patience of 2. To expedite model training, all models undergo finetuning using FP16 mixed precision and 2 gradient accumulation steps.

A.2.2 Experiment 2 - Lexical Training

We follow the setup from the previous experiment, but only slightly increase the learning rate to 1 x 10^3 in order to encourage more rapid adaptation to the new vocabulary.

A.2.3 Hardware Setup

We perform all our experiments with 8 NVIDIA A100s with 40 GiB vRAM.

Model Size	Hours
70M	0.65
160M	0.74
410M	1.2
1.4B	1.7

Table 5: Average GPU Hours For Full Finetuning

Model Size	Training Time Reduction (%)
70M	18.8
160M	21.1
410M	16.5
1.4B	21.5

Table 6: Δ compute time switching from standard full finetuning to lexical finetuning.

Model Size	# P	# Non-Embedding P
70M	70,426,624	18,915,328
160M	162,322,944	85,056,000
410M	405,334,016	302,311,424
1.4B	1,414,647,808	1,208,602,624

Table 7: Model Parameters (P), Available on HuggingFace.

A.3 Details on How to Reproduce PTP

We provide details on how to reproduce PTP in Algorithm 1.

A.4 Templates additions to `MISGENDERED`

To mimic real world pronoun declarations, each declaration is started with nominative, accusative, pronominal possessive, and reflexive pronouns.

Table 8 reflects selected additions from the `TANGO` dataset. `Det` represents the determiner position one may replace with ones like *the, a, these, those*. `Gen-dep, Gen-indep, reflex, nom` are all pronoun cases.

A.5 Example Generations

Table 4 example generations from the prompt *Skyler's pronouns are xe/xem/xir/xirself.*

Algorithm 1 Pronoun Tokenization Parity (PTP)

Input 1: LLM model
Input 2: LLM model's BPE tokenizer
Input 3: Defined list of neopronouns for PTP
Input 4: Dataset augmented with neopronouns
Method: Add special tokens for each neopronoun. Be sure to explicitly add 'Ġ' to the beginning of each token to indicate that it is a full, non-subword token space before the word, otherwise this will lead to incorrect model behavior, since a lack of 'Ġ' in BPE tokenization indicates a subword token.
Check: Check the tokenizer is working properly by checking the tokenized neopronoun, ensuring that you see 'Ġ' in its token. For example, tokenizing *xe* should result in ['Ġxe'] not ['Ġ', xe']. The latter will cause the LLM to incorrectly associate a space character with a neopronoun. This can be tested by checking next word transition probabilities from the space character.
Resize the LLM token embeddings to match vocabulary of tokenizer. Here is example code to do this with a model and tokenizer from `HuggingFace Transformers` Package [7].

```
#declare neopronoun tokens
arr_tokens = [
    'Ġxe', 'ĠXe',
    'Ġxem', 'ĠXem',
    'Ġxir', 'ĠXir',
    'Ġxirs', 'ĠXirs'
]

# add new tokens to the tokenizer, t
token_dict = {
'additional_special_tokens': arr_tokens
}
t.add_special_tokens(token_dict)

# update model, m, accordingly
m.resize_token_embeddings(len(tokenizer))
```

if `Lexical Finetuning` **then**
    Freeze all parameters besides the word token embeddings. Then proceed to finetune this lexical layer.
else
    Proceed with standard full finetuning
end if
Return Finetuned model, new PTP tokenizer
Evaluate using extended `MISGENDERED` framework

Case	Template
Nominative	before {name} came out, {det} {nom} identified differently. {name} is an indian filmmaker known for directing a movie, however {det} {nom} also directs television. {name} is a musician and dancer and {det} {nom} currently lives somewhere nearby.
Accusative	upon recognizing {name}, the fan asked {det} {acc} for an autograph. {name} owns the car so be sure to give the keys back to {det} {acc} afterwards. {name} needs new shoes so I can give my other shoes to {det} {acc}.
Pronominal Possessive	{name} was a participant in {det} {gen-indep} own musical. {name} is an author of children's fantasy, best known for {det} {gen-indep} book that won several awards. {name} is a singer-songwriter-producer based in boston and {det} {gen-indep} debut album was released in march.
Predicative Possessive	{name} was working in this area, so the phone is probably {det} {gen-dep}. the story of {name}'s resilience and triumph over adversity is {det} {gen-dep} to own. name} said to me that the larger slice of pizza was mine, and the smaller one was {det} {gen-dep}.
Reflexive	{name} will read the book by {det} {reflex}. {name} needs to be by {det} {reflex} sometimes. {name} often works alone by {det} {reflex}.

Table 8: Template additions to MISGENDERED (Hossain et al., 2023)

A.6 PTP Training Evaluation

We report cross entropy loss for the train and test across each model in Figure 6.

A.7 Downstream Evaluations

A.7.1 Setup

To confirm that our proposed techniques do not adversely affect downstream performance, we assess our models on three benchmarks for pronoun resolution and coreference resolution, logical reasoning , and knowledge retrieval respectively: WINOGRANDE (5-shot) (Sakaguchi et al., 2021), LOGIQA (5-shot) (Liu et al., 2021), and ARC-CHALLENGE (5-shot) (Clark et al., 2018). We utilize the LM evaluation harness [8] and discuss the results in the following subsections.

A.7.2 Results

We report our results in Table 9. For Winogrande, half of the models employing our methods either sustain or slightly boost performance, ranging from 0.08 to 0.24 points, likely due to improvement in pronoun disambiguation. For 410M and 1.4B, this boost is not observed. These base models slightly outperform our experiments, though the differences are marginal (1-2%) and insignificant.

For ARC, PTP and lexical finetuning either sustain or slightly improve baseline performance (1-

[8] https://github.com/EleutherAI/lm-evaluation-harness

Size	Version	Wino	ARC	LogiQA
70M	Base	$49.17_{1.41}$	$19.71_{1.16}$	$27.96_{1.76}$
	$T_{Orig} + M_{Full}$	$49.64_{1.41}$	$22.18_{1.21}$	$26.57_{1.73}$
	$T_{PTP} + M_{Base}$	$50.43_{1.41}$	$21.93_{1.21}$	$25.50_{1.71}$
	$T_{Orig} + M_{Lex}$	$50.51_{1.41}$	$\mathbf{23.72_{1.24}}$	$29.03_{1.78}$
	$T_{PTP} + M_{Lex}$	$\mathbf{50.99_{1.40}}$	$\mathbf{23.72_{1.24}}$	$\mathbf{29.49_{1.79}}$
160M	Base	$49.72_{1.41}$	$23.63_{1.24}$	$26.27_{1.73}$
	$T_{Orig} + M_{Full}$	$52.09_{1.40}$	$24.74_{1.26}$	$25.96_{1.72}$
	$T_{PTP} + M_{Base}$	$\mathbf{52.17_{1.40}}$	$24.15_{1.25}$	$27.04_{1.74}$
	$T_{Orig} + M_{Lex}$	$48.38_{1.40}$	$24.49_{1.26}$	$\mathbf{29.80_{1.79}}$
	$T_{PTP} + M_{Lex}$	$48.15_{1.40}$	$\mathbf{26.02_{1.28}}$	$29.49_{1.79}$
410M	Base	$\mathbf{54.85_{1.40}}$	$25.85_{1.28}$	$24.12_{1.68}$
	$T_{Orig} + M_{Full}$	$52.88_{1.40}$	$27.22_{1.30}$	$27.04_{1.74}$
	$T_{PTP} + M_{Base}$	$52.96_{1.40}$	$25.34_{1.27}$	$26.57_{1.73}$
	$T_{Orig} + M_{Lex}$	$51.46_{1.40}$	$25.26_{1.27}$	$\mathbf{27.80_{1.76}}$
	$T_{PTP} + M_{Lex}$	$51.46_{1.40}$	$\mathbf{27.39_{1.30}}$	$27.50_{1.75}$
1.4B	Base	$\mathbf{56.43_{1.39}}$	$\mathbf{32.17_{1.37}}$	$22.89_{1.65}$
	$T_{Orig} + M_{Full}$	$53.75_{1.40}$	$26.19_{1.28}$	$27.34_{1.75}$
	$T_{PTP} + M_{Base}$	$53.99_{1.40}$	$24.91_{1.26}$	$26.88_{1.74}$
	$T_{Orig} + M_{Lex}$	$52.72_{1.40}$	$30.03_{1.34}$	$\mathbf{28.73_{1.77}}$
	$T_{PTP} + M_{Lex}$	$52.72_{1.40}$	$28.75_{1.32}$	$28.57_{1.77}$

Table 9: Downstream Evaluations Across Model Size. Subscripts reflect standard deviations.

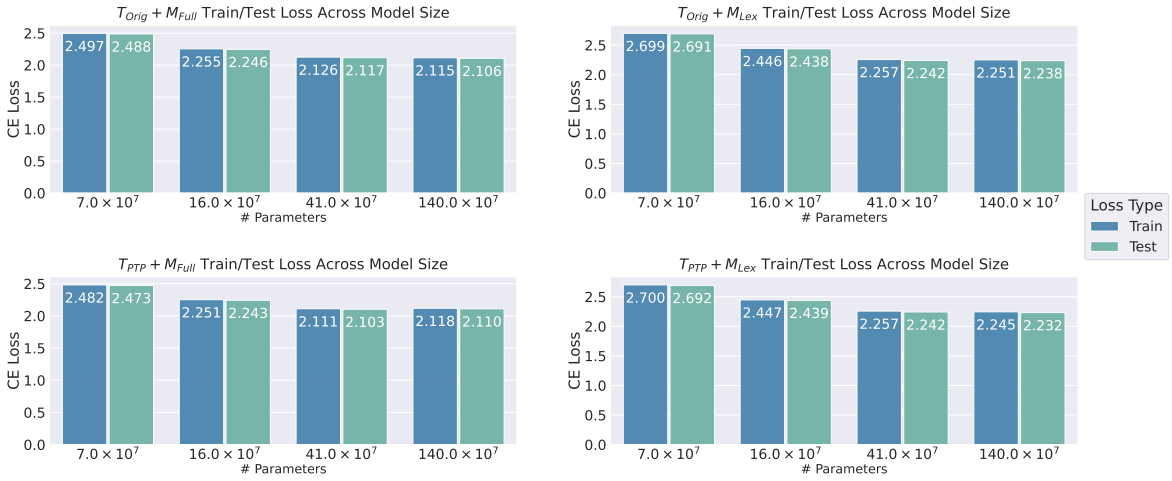

Figure 6: Reported Cross Entropy Loss for train/test across models.

2%) for most model sizes. For the 70M model, all lexical training outperforms full finetuning with original tokenization and the base model. We find this pattern consistent for 160M, and 410M. For the 1.4B model, the base model outperforms regular full finetuning with a 7% gap for full finetuning on original tokenization. In contrast, both lexical techniques outperform finetuning with both the original tokenizer and PTP. This finding indicates that combining PTP with lexical layer finetuning may be the best option for the highest pronoun gains while maintaining existing LLM capabilities.

For LogiQA, our methods either improve or are within the range of the baseline model. Namely, lexical finetuning corresponds to a good improvement over baseline. This finding is likely related to focused improvements in the LLM's lexical layers overall. Across all model sizes, both lexical training consistently outperforms finetuning without PTP and the base models. Our findings suggest that lexical layer finetuning, with or without vocabulary expansion, does not harm the model's downstream performance on LogiQA compared to regular finetuning or the base models.

A.8 Ablations

Table 10 provides results across all data splits for the 70M model. Table 11 provides results across model sizes for the 10% data resource ablation, so as to best mimic real-world low-resource circumstances.

B Ablations Across Size and Data Resource

Model	Pronoun Consistency (↑)			Case Error (↓)			Inject Error (↓)		
	He	She	Xe	He	She	Xe	He	She	Xe
$T_{ORIG} + M_{BASE}$	$96.82_{0.79}$	$71.59_{2.00}$	$0.67_{0.33}$	$8.26_{1.23}$	$24.36_{1.92}$	$78.56_{1.82}$	$23.85_{1.87}$	$16.92_{1.67}$	$85.03_{1.59}$
$T_{ORIG} + M_{FULL}$	$89.64_{1.33}$	$86.05_{1.54}$	$14.46_{1.56}$	$11.74_{1.44}$	$22.41_{1.82}$	$59.95_{2.18}$	$23.95_{1.90}$	$16.77_{1.69}$	$89.49_{1.36}$
$T_{ORIG} + M_{LEX}$	$86.46_{1.54}$	$72.87_{1.97}$	$16.77_{1.67}$	$18.51_{1.74}$	$33.79_{2.10}$	$70.51_{2.00}$	$28.97_{2.03}$	$23.18_{1.87}$	$65.44_{2.13}$
$T_{PTP} + M_{FULL}$	$94.77_{0.97}$	$83.49_{1.64}$	$37.79_{2.18}$	$9.69_{1.31}$	$29.28_{2.00}$	$56.92_{2.21}$	$27.79_{1.97}$	$20.97_{1.82}$	$27.03_{1.92}$
$T_{PTP-B} + M_{FULL}$	$96.21_{0.85}$	$80.72_{1.77}$	$24.36_{1.90}$	$9.49_{1.31}$	$31.33_{2.05}$	$61.90_{2.18}$	$28.26_{2.03}$	$20.56_{1.77}$	$25.95_{1.95}$
$T_{PTP} + M_{LEX}$	$84.97_{1.56}$	$72.21_{1.97}$	$53.59_{2.23}$	$18.15_{1.69}$	$33.03_{2.10}$	$60.46_{2.15}$	$25.79_{1.95}$	$21.85_{1.85}$	$34.77_{2.10}$
$T_{PTP-B} + M_{LEX}$	$83.28_{1.64}$	$74.31_{1.97}$	$42.97_{2.23}$	$16.10_{1.64}$	$33.33_{2.08}$	$57.74_{2.18}$	$24.31_{1.90}$	$20.21_{1.79}$	$32.05_{2.08}$

(a) Data Split=10

Model	Consistency (↑)			Case Error (↓)			Inject Error (↓)		
	He	She	Xe	He	She	Xe	He	She	Xe
$T_{ORIG} + M_{BASE}$	$96.82_{0.77}$	$71.59_{2.03}$	$0.67_{0.36}$	$8.26_{1.23}$	$24.36_{1.90}$	$78.56_{1.85}$	$23.85_{1.87}$	$16.92_{1.67}$	$85.03_{1.59}$
$T_{ORIG} + M_{FULL}$	$93.23_{1.10}$	$81.44_{1.77}$	$13.74_{1.56}$	$11.69_{1.44}$	$24.97_{1.92}$	$58.77_{2.15}$	$27.08_{1.95}$	$18.00_{1.72}$	$87.28_{1.46}$
$T_{ORIG} + M_{LEX}$	$86.05_{1.51}$	$73.33_{1.95}$	$18.10_{1.69}$	$17.03_{1.67}$	$32.00_{2.10}$	$71.38_{2.00}$	$27.59_{1.97}$	$19.90_{1.74}$	$67.54_{2.10}$
$T_{PTP} + M_{FULL}$	$96.51_{0.82}$	$88.56_{1.38}$	$35.95_{2.10}$	$11.59_{1.41}$	$32.05_{2.05}$	$47.54_{2.21}$	$25.28_{1.97}$	$19.18_{1.77}$	$33.85_{2.05}$
$T_{PTP-B} + M_{FULL}$	$95.28_{0.92}$	$87.33_{1.46}$	$18.51_{1.69}$	$9.95_{1.33}$	$30.72_{2.00}$	$48.41_{2.18}$	$26.87_{1.92}$	$19.54_{1.74}$	$34.00_{2.10}$
$T_{PTP} + M_{LEX}$	$82.21_{1.69}$	$70.87_{2.03}$	$48.00_{2.23}$	$15.44_{1.64}$	$31.59_{2.05}$	$59.23_{2.18}$	$30.10_{2.03}$	$23.69_{1.87}$	$34.92_{2.10}$
$T_{PTP-B} + M_{LEX}$	$83.18_{1.67}$	$70.05_{2.00}$	$32.41_{2.03}$	$15.28_{1.59}$	$32.92_{2.08}$	$57.95_{2.21}$	$30.05_{2.05}$	$22.62_{1.87}$	$34.00_{2.08}$

(b) Data Split=20

Model	Consistency (↑)			Case Error (↓)			Inject Error (↓)		
	He	She	Xe	He	She	Xe	He	She	Xe
$T_{ORIG} + M_{BASE}$	$96.82_{0.77}$	$71.59_{2.00}$	$0.67_{0.36}$	$8.26_{1.23}$	$24.36_{1.90}$	$78.56_{1.82}$	$23.85_{1.87}$	$16.92_{1.64}$	$85.03_{1.59}$
$T_{ORIG} + M_{FULL}$	$91.28_{1.26}$	$85.23_{1.59}$	$13.85_{1.51}$	$12.87_{1.46}$	$21.90_{1.82}$	$60.62_{2.18}$	$24.56_{1.90}$	$19.08_{1.72}$	$87.03_{1.49}$
$T_{ORIG} + M_{LEX}$	$80.10_{1.77}$	$64.67_{2.10}$	$18.46_{1.74}$	$22.62_{1.85}$	$34.56_{2.10}$	$68.87_{2.08}$	$29.18_{2.00}$	$24.26_{1.92}$	$66.56_{2.10}$
$T_{PTP} + M_{FULL}$	$95.79_{0.90}$	$87.69_{1.44}$	$32.41_{2.08}$	$13.44_{1.51}$	$28.51_{2.00}$	$46.92_{2.18}$	$23.18_{1.90}$	$19.69_{1.74}$	$34.41_{2.13}$
$T_{PTP-B} + M_{FULL}$	$90.87_{1.28}$	$84.41_{1.56}$	$12.56_{1.49}$	$10.46_{1.36}$	$30.00_{2.05}$	$49.33_{2.23}$	$25.49_{1.95}$	$19.13_{1.74}$	$26.00_{1.97}$
$T_{PTP} + M_{LEX}$	$81.23_{1.72}$	$62.00_{2.15}$	$45.49_{2.21}$	$19.64_{1.77}$	$35.74_{2.15}$	$55.49_{2.18}$	$26.77_{1.95}$	$20.92_{1.82}$	$31.44_{2.05}$
$T_{PTP-B} + M_{LEX}$	$84.87_{1.59}$	$69.33_{2.08}$	$48.26_{2.23}$	$20.72_{1.79}$	$35.79_{2.08}$	$53.33_{2.23}$	$27.69_{2.00}$	$20.97_{1.79}$	$33.33_{2.10}$

(c) Data Split=30

Model	Consistency (↑)			Case Error (↓)			Inject Error (↓)		
	He	She	Xe	He	She	Xe	He	She	Xe
$T_{ORIG} + M_{BASE}$	$96.82_{0.77}$	$71.59_{2.00}$	$0.67_{0.38}$	$8.26_{1.23}$	$24.36_{1.87}$	$78.56_{1.79}$	$23.85_{1.87}$	$16.92_{1.67}$	$85.03_{1.59}$
$T_{ORIG} + M_{FULL}$	$92.41_{1.18}$	$79.33_{1.79}$	$9.95_{1.31}$	$14.97_{1.56}$	$21.28_{1.79}$	$60.31_{2.15}$	$25.38_{1.92}$	$19.64_{1.79}$	$85.54_{1.56}$
$T_{ORIG} + M_{LEX}$	$82.15_{1.72}$	$65.23_{2.15}$	$18.21_{1.72}$	$24.00_{1.92}$	$33.03_{2.05}$	$67.38_{2.08}$	$31.08_{2.08}$	$22.56_{1.85}$	$68.15_{2.08}$
$T_{PTP} + M_{FULL}$	$96.00_{0.87}$	$86.41_{1.56}$	$26.82_{1.95}$	$15.33_{1.59}$	$32.77_{2.08}$	$47.38_{2.21}$	$25.13_{1.95}$	$20.00_{1.72}$	$33.90_{2.13}$
$T_{PTP-B} + M_{FULL}$	$96.67_{0.79}$	$86.15_{1.49}$	$11.69_{1.44}$	$8.72_{1.23}$	$32.00_{2.05}$	$48.21_{2.23}$	$23.44_{1.85}$	$20.26_{1.77}$	$33.95_{2.10}$
$T_{PTP} + M_{LEX}$	$85.33_{1.56}$	$61.49_{2.15}$	$48.41_{2.26}$	$22.15_{1.85}$	$37.74_{2.13}$	$53.59_{2.15}$	$28.97_{2.03}$	$21.64_{1.79}$	$33.18_{2.10}$
$T_{PTP-B} + M_{LEX}$	$84.92_{1.59}$	$62.00_{2.21}$	$41.44_{2.21}$	$21.69_{1.82}$	$38.26_{2.15}$	$53.08_{2.21}$	$28.92_{2.00}$	$22.87_{1.87}$	$33.08_{2.10}$

(d) Data Split=40

Model	Consistency (↑)			Case Error (↓)			Inject Error (↓)		
	He	She	Xe	He	She	Xe	He	She	Xe
$T_{ORIG} + M_{BASE}$	$96.82_{0.79}$	$71.59_{2.00}$	$0.67_{0.36}$	$8.26_{1.23}$	$24.36_{1.87}$	$78.56_{1.85}$	$23.85_{1.90}$	$16.92_{1.64}$	$85.03_{1.56}$
$T_{ORIG} + M_{FULL}$	$93.44_{1.08}$	$85.23_{1.54}$	$14.05_{1.54}$	$9.59_{1.33}$	$23.08_{1.87}$	$59.28_{2.18}$	$26.00_{1.97}$	$19.79_{1.79}$	$86.10_{1.54}$
$T_{ORIG} + M_{LEX}$	$83.38_{1.67}$	$65.13_{2.13}$	$18.46_{1.69}$	$20.51_{1.79}$	$36.82_{2.13}$	$69.54_{2.05}$	$28.72_{2.05}$	$19.03_{1.72}$	$71.18_{2.03}$
$T_{PTP} + M_{FULL}$	$96.00_{0.87}$	$88.92_{1.36}$	$26.67_{1.97}$	$13.64_{1.54}$	$31.64_{2.08}$	$45.90_{2.23}$	$24.36_{1.90}$	$21.69_{1.87}$	$35.90_{2.10}$
$T_{PTP-B} + M_{FULL}$	$95.03_{0.97}$	$87.23_{1.51}$	$16.10_{1.64}$	$10.97_{1.38}$	$33.08_{2.05}$	$48.36_{2.21}$	$29.49_{2.03}$	$21.59_{1.82}$	$37.90_{2.15}$
$T_{PTP} + M_{LEX}$	$77.54_{1.85}$	$58.15_{2.23}$	$58.41_{2.18}$	$21.64_{1.85}$	$37.74_{2.18}$	$50.87_{2.23}$	$29.13_{2.03}$	$19.74_{1.82}$	$31.54_{2.03}$
$T_{PTP-B} + M_{LEX}$	$81.54_{1.72}$	$64.41_{2.13}$	$49.28_{2.21}$	$19.95_{1.77}$	$37.85_{2.13}$	$52.67_{2.21}$	$26.77_{1.92}$	$22.41_{1.82}$	$30.51_{2.05}$

(e) Data Split=50

Table 10: 70M Model Results Across Data Splits

Model	Consistency (↑)			Case Error (↓)			Inject Error (↓)		
	He	She	Xe	He	She	Xe	He	She	Xe
$T_{ORIG} + M_{BASE}$	$79.95_{1.77}$	$76.46_{1.87}$	$0.00_{0.00}$	$4.05_{0.85}$	$10.87_{1.36}$	$80.00_{1.74}$	$8.72_{1.26}$	$6.46_{1.08}$	$95.38_{0.95}$
$T_{ORIG} + M_{FULL}$	$78.87_{1.79}$	$61.49_{2.15}$	$15.59_{1.64}$	$11.23_{1.38}$	$20.21_{1.77}$	$48.92_{2.21}$	$19.44_{1.74}$	$20.31_{1.79}$	$69.18_{2.03}$
$T_{ORIG} + M_{LEX}$	$77.28_{1.82}$	$70.05_{2.03}$	$20.00_{1.77}$	$12.56_{1.46}$	$23.90_{1.90}$	$57.59_{2.18}$	$20.21_{1.79}$	$16.87_{1.67}$	$78.26_{1.85}$
$T_{PTP} + M_{FULL}$	$80.21_{1.79}$	$64.92_{2.08}$	$30.92_{2.03}$	$6.21_{1.05}$	$23.59_{1.90}$	$56.26_{2.18}$	$22.00_{1.87}$	$18.15_{1.72}$	$14.72_{1.62}$
$T_{PTP-B} + M_{FULL}$	$79.13_{1.79}$	$65.79_{2.08}$	$9.74_{1.33}$	$8.26_{1.21}$	$22.51_{1.87}$	$59.85_{2.15}$	$20.87_{1.79}$	$21.03_{1.82}$	$25.28_{1.92}$
$T_{PTP} + M_{LEX}$	$78.51_{1.82}$	$60.77_{2.21}$	$51.79_{2.23}$	$12.10_{1.41}$	$27.64_{1.97}$	$46.36_{2.23}$	$19.13_{1.74}$	$14.77_{1.62}$	$31.44_{2.03}$
$T_{PTP-B} + M_{LEX}$	$81.23_{1.72}$	$60.46_{2.15}$	$53.64_{2.18}$	$13.38_{1.51}$	$29.18_{2.00}$	$47.49_{2.21}$	$17.49_{1.69}$	$16.41_{1.67}$	$25.13_{1.95}$

(a) 160M Parameter Model Results

Model	Consistency (↑)			Case Error (↓)			Inject Error (↓)		
	He	She	Xe	He	She	Xe	He	She	Xe
$T_{ORIG} + M_{BASE}$	$72.82_{1.95}$	$55.85_{2.18}$	$0.05_{0.08}$	$76.72_{1.87}$	$66.26_{2.13}$	$20.00_{1.74}$	$4.15_{0.87}$	$3.49_{0.82}$	$89.85_{1.36}$
$T_{ORIG} + M_{FULL}$	$79.03_{1.82}$	$42.10_{2.21}$	$18.36_{1.72}$	$74.41_{1.92}$	$61.69_{2.15}$	$20.05_{1.77}$	$12.82_{1.46}$	$19.79_{1.74}$	$56.62_{2.15}$
$T_{ORIG} + M_{LEX}$	$69.85_{2.03}$	$42.10_{2.18}$	$19.85_{1.77}$	$76.51_{1.92}$	$65.33_{2.10}$	$20.97_{1.79}$	$16.97_{1.67}$	$11.79_{1.44}$	$54.51_{2.23}$
$T_{PTP} + M_{FULL}$	$63.64_{2.13}$	$48.21_{2.23}$	$56.21_{2.15}$	$73.69_{1.92}$	$62.00_{2.15}$	$40.56_{2.23}$	$14.36_{1.56}$	$14.97_{1.56}$	$20.67_{1.77}$
$T_{PTP-B} + M_{FULL}$	$82.31_{1.69}$	$48.82_{2.21}$	$19.03_{1.77}$	$73.23_{1.95}$	$61.23_{2.15}$	$41.69_{2.21}$	$11.54_{1.41}$	$17.38_{1.69}$	$26.21_{1.92}$
$T_{PTP} + M_{LEX}$	$68.31_{2.05}$	$45.74_{2.21}$	$40.62_{2.21}$	$76.51_{1.90}$	$64.15_{2.15}$	$47.79_{2.18}$	$16.72_{1.64}$	$13.38_{1.51}$	$13.59_{1.54}$
$T_{PTP-B} + M_{LEX}$	$69.44_{2.05}$	$35.59_{2.15}$	$49.18_{2.23}$	$77.49_{1.87}$	$65.49_{2.10}$	$48.67_{2.23}$	$17.74_{1.69}$	$17.44_{1.67}$	$12.26_{1.46}$

(b) 410m Parameter Model Results

Model	Consistency (↑)			Case Error (↓)			Inject Error (↓)		
	He	She	Xe	He	She	Xe	He	She	Xe
$T_{ORIG} + M_{BASE}$	$78.46_{1.82}$	$66.56_{2.08}$	$0.26_{0.23}$	$3.54_{0.85}$	$3.03_{0.77}$	$76.00_{1.90}$	$3.69_{0.85}$	$3.44_{0.79}$	$92.77_{1.15}$
$T_{ORIG} + M_{FULL}$	$76.72_{1.87}$	$58.72_{2.18}$	$17.90_{1.72}$	$8.10_{1.23}$	$25.18_{1.92}$	$36.46_{2.15}$	$24.72_{1.90}$	$24.56_{1.92}$	$36.31_{2.13}$
$T_{ORIG} + M_{LEX}$	$80.05_{1.77}$	$56.00_{2.18}$	$15.49_{1.59}$	$5.64_{1.03}$	$17.90_{1.72}$	$40.82_{2.21}$	$16.62_{1.69}$	$35.49_{2.13}$	$55.23_{2.21}$
$T_{PTP} + M_{FULL}$	$84.72_{1.64}$	$56.46_{2.21}$	$44.97_{2.21}$	$4.56_{0.92}$	$20.31_{1.79}$	$44.92_{2.21}$	$24.10_{1.90}$	$18.00_{1.69}$	$20.05_{1.77}$
$T_{PTP-B} + M_{FULL}$	$71.90_{2.00}$	$53.95_{2.21}$	$35.69_{2.13}$	$8.41_{1.26}$	$18.00_{1.72}$	$40.10_{2.18}$	$19.13_{1.79}$	$22.31_{1.82}$	$18.51_{1.72}$
$T_{PTP} + M_{LEX}$	$76.77_{1.90}$	$44.62_{2.21}$	$36.26_{2.10}$	$2.56_{0.69}$	$18.62_{1.72}$	$31.54_{2.08}$	$12.05_{1.44}$	$24.56_{1.90}$	$19.33_{1.74}$
$T_{PTP-B} + M_{LEX}$	$79.74_{1.82}$	$57.85_{2.18}$	$35.64_{2.13}$	$4.67_{0.92}$	$14.62_{1.56}$	$33.13_{2.10}$	$20.10_{1.77}$	$26.72_{1.97}$	$27.23_{1.97}$

(c) 1.4B Parameter Model Results

Table 11: Model Size Comparisons at Data Split=10