Appendix

A. Comparison with Related Benchmarks

We highlight the following differences between our notion and evaluation of representational bias in pretrained LMs with recent work in this direction:

1. The Sentence Encoder Association Test (May et al., 2019) extend WEAT to sentence encoders by creating artificial sentences using templates of the form *"This is [target]"* and *"They are [attribute]"*. SEAT is primarily a method to measure bias in contextual embeddings and does not extend to generation.

2. StereoSet (Nadeem et al., 2020) defines a set of attributes spanning professions, race, and religion from Wikipedia before asking a crowdworker to write attribute terms that correspond to stereotypical, anti-stereotypical and unrelated associations of the target term. We believe that StereoSet is a valuable resource with well-defined tests for both intrasentence and intersentence stereotypical associations and we report results on this benchmark. However, there is a lack of diversity regarding the contexts chosen, and as a result, it is unable to clearly measure fine-grained context and bias associations in pretrained LMs.

3. In Sheng et al. (2019), the authors choose a set of contexts and obtain the completed sentences via pretrained LMs before measuring differences in *regard* across generated sentences from different social contexts. Again, they suffer in the diversity of contexts since they begin with a small set of bias terms (e.g., *man/woman*) and use simple placeholder templates (e.g., *"The woman worked as"*, *"The man was known for"*). This does not allow testing over diverse templates which implies an inability to disentangle fine-grained context and bias associations in pretrained LMs.

B. Benchmarks for Measuring Bias

B.1. Collecting Diverse Contexts

To accurately benchmark LMs for both bias and context associations, it is also important to use *diverse contexts* beyond simple templates used in prior work. Specifically, the Sentence Encoder Association Test (May et al., 2019), StereoSet (Nadeem et al., 2020)), and templates in Sheng et al. (2019) are all based on combining bias terms (e.g., gender and race terms) and attributes (e.g., professions) with simple placeholder templates (e.g., *The woman worked as*, *The man was known for*). Diverse contexts found in naturally occurring text corpora contain important context associations to accurately benchmark whether the new LM can still accurately generate realistic text, while also ensuring that the biases in the new LM are tested in rich real-world contexts.

To achieve this, we collect a large set of $16,338$ diverse contexts from 5 real-world text corpora. Our text corpora originate from the following five sources: 1) **WikiText-2** (Merity et al., 2017), a dataset of formally written Wikipedia articles (we only use the first 10% of WikiText-2 which we found to be sufficient to capture formally written text), 2) **Stanford Sentiment Treebank** (Socher et al., 2013), a collection of $10,000$ polarized written movie reviews, 3) **Reddit** data collected from discussion forums related to politics, electronics, and relationships, 4) **MELD** (Poria et al., 2019), a large-scale multimodal multi-party emotional dialog dataset collected from the TV-series Friends, and 5) **POM** (Park et al., 2014), a dataset of spoken review videos collected across $1,000$ individuals spanning multiple topics. These datasets have been the subject of recent research in language understanding (Merity et al., 2017; Liu et al., 2019; Wang et al., 2019) and multimodal human language (Liang et al., 2018). Table 9 summarizes these datasets. In Table 9, we give some examples of the diverse templates that occur naturally across various individuals, settings, and in both written and spoken text. To measure language model performance, we randomly choose 50 contexts for each bias class. For measuring bias, we sample 100 contexts for each bias class and generate swapped context pairs.

C. Experimental Details

C.1. Implementation Details

All models and analysis were done in Python. The pretrained GPT-2 model was implemented using Hugging Face (Wolf et al., 2020) (website: `https://huggingface.co`, GitHub: `https://github.com/huggingface`).

Table 9. Comparison of the various datasets used to find diverse contexts for measuring social biases in language models. Length represents the average length measured by the number of words in a sentence. Words in italics indicate the words used to estimating the binary gender or multiclass religion subspaces, e.g. (*man*, *woman*), (*jewish*, *christian*, *muslim*). This demonstrates the variety in our diverse contexts in terms of topics, formality, and spoken/written text.

Dataset	Type	Topics	Formality	Length	Samples
WikiText-2	written	everything	formal	24.0	"the mailing contained information about their history and advised people to read several books, which primarily focused on {*jewish*/*christian*/*muslim*} history"
SST	written	movie reviews	informal	19.2	"{*his*/*her*} fans walked out muttering words like horrible and terrible, but had so much fun dissing the film that they didn't mind the ticket cost."
Reddit	written	politics, electronics, relationships	informal	13.6	"roommate cut my hair without my consent, ended up cutting {*himself* /*herself* } and is threatening to call the police on me"
MELD	spoken	comedy TV-series	informal	8.1	"that's the kind of strength that I want in the {*man*/*woman*} I love!"
POM	spoken	opinion videos	informal	16.0	"and {*his*/*her*} family is, like, incredibly confused"

C.2. Efficient Implementation by Caching

Finally, we note that naive implementation of our algorithm might seem to require repeated forward passes corresponding to autoregressively feeding output tokens into the prior conditioning text. However, practical efficient implementations of the Transformer (Wolf et al., 2020) use a cached context embedding $f(c_{t-1})$ to generate w_t, given w_{t-1}. This recurrent interpretation of a transformer can be summarized as:

$$o_t, H_t = \text{LM}(w_{t-1}, f(c_{t-1})) \tag{9}$$

where the encoded context $f(c_{t-1})$ denotes the history consisting of the key-value pairs from the past, i.e., $f(c_{t-1}) = [(K_{t-1}^{(1)}, V_{t-1}^{(1)}), ..., (K_{t-1}^{(l)}, V_{t-1}^{(l)})]$ where $(K_{t-1}^{(1)}, V_{t-1}^{(1)})$ corresponds to the key-value pairs from the i-th Transformer layer generated from time steps 0 to $t-1$.

Given a linear transformation W that maps the logit vector o_t to a vector of vocabulary size, x_t is then sampled as $x_t \sim p_t = \text{Softmax}(W o_t)$. This allows for efficient language generation without repeated forward passes corresponding to the prior conditioning tokens $w_0, ..., w_{t-1}$ (see Dathathri et al. (2019) for more details).

C.3. Hyperparameters

We performed a small hyperparameter search over the ranges in Table 10 and Table 11. By choosing the better performing model, we selected the resulting hyperparameters as shown in bold in Table 10 and Table 11. To learn the bias SVM classifier, we selected the best hyperparamter choosing the best performance on the validation dataset. During debiasing, we selected the best hyperparamter that achieved the best performance-fairness tradeoff (largest area under the performance-fairness curve).

C.4. Model Parameters

SVM model has 2307 parameters $(768 * 3 + 3)$ and small GPT-2 has 124 million parameters. The nullspace matrix P has $589,000$ parameters (768 $*$ 768).

C.5. Training Resources and Time

All experiments were conducted on a Tesla P40 Ti GPU with 22 GB memory. We analyze the additional time and space complexity of our approach. The main bottleneck lies in the preprocessing phase which can then be amortized over multiple inference runs in mitigating biases. The preprocessing phase takes 740 seconds and 1470 MiB memory. For inference pass, it takes 102 seconds to load and initialize the model and the tokenizer. It takes 1.21 seconds and 1231 MiB memory to generate a single sentence an average length of 25 as compared to 1.12 seconds and 1181 MiB memory for the original GPT-2 language model. Therefore, our A-INLP approach incurs negligible additional time and space complexity during inference.

Table 10. Model hyperparameter configurations for experiments in mitigating gender biases. The list shows all hyperparameters tested with the final selected hyperparameter (based on best validation set performance) in bold.

Model	Parameter	Value
Bias Sensitive Tokens/Context	word embedding	**GloVe embedding**, GPT-2 embedding
	number of definitional bias pairs	$1, 3, 5, \mathbf{10}, 15$
	number of components of subspace	$1, 2, \mathbf{3}, 5, 10$
	number of bias sensitive token	$50, 100, 200, \mathbf{500}, 1000$
Null Space Projection	size of the dataset	$3000, 4500, \mathbf{6000}, 7500$
	number of iteration	$40, 50, 60, 70, \mathbf{80}, 90$
	dropout	$\mathbf{0}, 0.1, 0.2, 0.3$
SVM	C	$0.1, 0.5, \mathbf{1}, 2, 3, 5, 10$
	penalty	$\ell_1, \boldsymbol{\ell_2}$
	loss	hinge, **squared_hinge**
	optimization problem	dual, **primal**
	iteration	$500, \mathbf{1000}, 2000, 4000, 5000$
A-INLP	α	$0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9$
GPT-2	maximum length	$20, 25, \mathbf{30}, 35, 40$
	no repeat ngram size	$0, 1, 2, \mathbf{3}, 4, 5$
	repetition penalty	$1, 1.1, 1.2, 1.3, 1.4, \mathbf{1.5}, 1.6$
	temperature	$\mathbf{1}, 1.1, 1.2, 1.3, 1.4, 1.5$

D. Additional Results

D.1. Identifying Bias-Sensitive Tokens

To identify bias sensitive tokens from the whole vocabulary, we first estimate a bias subspace using several pre-defined bias pairs, such as *she* and *he* for gender, *jew*, *christian*, and *muslim* for religion (see Table 12 for the exact word pairs/triplets used). With multiple pairs, we can calculate the difference vectors of these pairs and apply PCA to obtain a bias subspace of token embedding. Following Manzini et al. (2019), formally, given defining sets of word embeddings $D_1, D_2, ..., D_n$, let the mean of the defining set i be $\mu_i = \frac{1}{|D_i|}\sum_{\mathbf{w}\in D_i}\mathbf{w}$, where $\mathbf{w}$ is the word embedding of w. Then the bias subspace B is given by the first k components of principal component analysis (PCA) on B:

$$B_k = \mathbf{PCA}\left(\bigcup_{i=1}^{n}\bigcup_{w\in D_i}\mathbf{w} - \mu_i\right) \tag{10}$$

We can calculate the projection of a new token embedding $\mathbf{w}'$ onto this subspace: $\text{proj}_{B_k}(\mathbf{w}') = \sum_{\mathbf{b}\in B_k}\mathbf{b}^\top\mathbf{w}'$. The projection value reflects the extent of bias and we can use it to identify bias sensitive tokens.

We test this algorithm using both GloVe word embeddings and GPT-2 context embedding. We find the subspace of GloVe embeddings is much more accurate than the GPT-2 embeddings, especially for religion. In Table 13, we provide top 100 biased tokens for each class in glove embedding. We also show the top 100 biased tokens in GPT-2 embedding in Table 14. Surprisingly, we find that several stop words have large projection values onto the male subspace, so we removed these stop words. Aside from these stop words, we found that many of the learned words very negatively stereotype certain genders and religions (especially for the female gender and Muslim religion).

D.2. Learning a Bias Classifier

Data collection: To obtain the nullspace of the bias classifier, we collect data from both simple templates from Sheng et al. (2019) and diverse sentences from real corpus as described in Appendix B. For the simple templates, we replace the *XYZ* placeholder (e.g., *The XYZ was known for*) with bias definitional tokens in Table 12. For experiments using diverse context, we first define a bias subspace and identify bias sensitive tokens. Then, we contextualize these bias sensitive tokens into bias sensitive contexts by collecting sentences which contain these bias sensitive tokens from real-world corpus (Appendix B). We remove sentences containing bias sensitive tokens across multiple classes and also remove sentences with less than 5 tokens. We randomly choose a subsequence of the full sentences as the context.

For experiments studying gender bias, we found a large amount of sentences containing gender sensitive tokens such as *his* and *her*. We randomly collect $15,162$ context samples in total. For experiments studying religion bias, the related sentences

Table 11. Model hyperparameter configurations for experiments in mitigating religion biases. The list shows all hyperparameters tested with the final selected hyperparameter (based on best validation set performance) in bold.

Model	Parameter	Value
Bias Sensitive Tokens/Context	word embedding	**GloVe embedding**, GPT-2 embedding
	number of definitional bias pairs	$1, 3, \mathbf{6}, 10, 15$
	number of components of subspace	$\mathbf{1}, 2, 3, 6, 10$
	number of bias sensitive token	$50, 100, 200, \mathbf{500}, 1000$
Null Space Projection	size of the dataset	$3000, 4500, \mathbf{6000}, 7500$
	number of iteration	$40, \mathbf{50}, 60, 70, 80, 90$
	dropout	$\mathbf{0}, 0.1, 0.2, 0.3$
SVM	C	$0.1, 0.5, \mathbf{1}, 2, 3, 5, 10$
	penalty	$\ell_1, \boldsymbol{\ell_2}$
	loss	hinge, **squared_hinge**
	optimization problem	dual, **primal**
	iteration	$500, 1000, \mathbf{2000}, 4000, 5000$
A-INLP	α	$0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9$
GPT-2	maximum length	$20, 25, \mathbf{30}, 35, 40$
	no repeat ngram size	$0, 1, 2, \mathbf{3}, 4, 5$
	repetition penalty	$1, 1.1, 1.2, 1.3, 1.4, \mathbf{1.5}, 1.6$
	temperature	$\mathbf{1}, 1.1, 1.2, 1.3, 1.4, 1.5$

Table 12. Definitional pairs used to estimate the bias subspace for gender and religion.

Class	pairs
Gender	*(woman, man), (girl, boy), (she, he), (mother, father), (daughter, son), (gal, guy), (female, male), (her, his), (herself, himself), (Mary, John)*
Religion	*(jewish, christian, muslim), (jews, christians, muslims), (torah, bible, quran), (synagogue, church, mosque), (rabbi, priest, imam), (judaism, christianity, islam)*

are much more rare. We obtain $1,176$ context samples from the corpus in total and nearly half of these samples contain *church* which indicates a single religion class *christian*. In order to increase the number of training samples as well as match to partial input contexts that are usually input to GPT-2, we supplement our contexts with several partial subsequences.

Another way to collect bias sensitive context is to define a context subspace via several definitional context pairs using the method proposed in Liang et al. (2020); May et al. (2019), and then collect contexts according to their projection onto this *context subspace*. However, we find that compared to a token-level subspace, context-level subspaces are much harder to estimate and give results with higher variance.

Overall, this data collection process results in $6,000$ context samples for our dataset split into $2,940$ training samples, $1,260$ validation samples and $1,800$ test samples.

Training the bias classifier: We train a linear SVM with ℓ_2 penalty and squared hinge loss as our bias classifier. Both gender and religion have three classes. For gender, we iteratively train 80 classifiers. For religion, we iteratively train 50 classifiers. The accuracy of the classifier is around 33% when we finish our algorithm, which means after the nullspace projection, the context embedding cannot be classified with respect to the bias attributes and thus does not contain distinguishable bias information.

D.3. Local and Global Bias

In this section we provide examples of more metrics and results for measuring and mitigating bias via local and global metrics.

Local metrics for fairness: Consider the generation of word w_t given a context $c_{t-1}^{(1)}$ describing the first social group (e.g., male individual). Change the context to $c_{t-1}^{(2)}$ such that it describes the second social group (e.g., female individual), and vice-versa. To measure local biases across the vocabulary, we use a suitable f-divergence between the probability distributions predicted by the LM conditioned on both counterfactual contexts. Computing the f-divergence has a nice

Table 13. Top 100 biased tokens for each social group as obtained using the GloVe embedding subspace. We find that many of the learned bias words very negatively stereotype certain genders and religions (especially for the female gender and Muslim religion).

Class	Attribute	Tokens
Gender	Male	*himself, john, his, paul, he, sir, man, manny, guy, arsene, drafted, trevor, chairman, david, dawkins, colonel, elway, capt, successor, captain, mike, drummer, ratzinger, danny, joe, emmanuel, aaron, dirkxin, tito, mitre, andrew, godfather, manuel, goodfellas, phil, jonny, baron, bernanke, ballmer, spokesman, richard, alan, brian, general, teilhard, jimbo, jim, rangers, karl, scorsese, stephen, king, peter, belichick, amir, dave, him, hagee, tim, qb, nick, lew, muhammad, bankster, kevin, sabean, ben, heyman, theo, genius, jon, rudy, schalk, englishman, henchman, nimrod, greg, buckethead, son, batista, steve, forefather, elazar, daniel, preached, luke, andy, tackle, malthus, reginald, roy, chief, walter, piltdown, shogun, daoud, punter, mr, johnny*
	Female	*ftv, nichole, sassy, menstruating, ballerina, goddess, pregnant, marie, lactating, diva, madeline, songstress, xoxo, engelbreit, tiana, elina, temptress, preggy, lingerie, seductress, hecate, sapphic, kayla, lenora, latina, alena, fishnets, motherhood, miyu, authoress, lactation, sophia, busty, herstory, czarina, bewitching, curvy, nightgown, helene, alumna, dowager, preggers, malissa, princess, adelia, actress, renee, cecelia, nympho, christina, katheryn, nubile, vixen, corset, madelyn, squirting, popova, dildoing, miscarry, heidi, lesbo, lillian, sophie, stacie, erika, louisa, pregant, addie, pregnancy, nicole, annabelle, whorish, samantha, heroine, adeline, linnea, milf, buxom, mikayla, kristine, louise, katelynn, housewife, bra, sqirting, trimester, johanna, femjoy, breastfeeding, hallie, elise, witchy, angelica, kristina, katarina, nadya, alya, slutty, moms, alyssa*
Religion	Jewish	*rabbinical, sephardic, rabbinic, hasidic, judaism, shabbat, kashrut, reconstructionist, sephardi, menorah, midrash, jewishness, latkes, halakha, halakhic, bnei, pesach, torah, rabbinate, kabbalistic, talmudic, rabbis, tikkun, hillel, lubavitch, judaica, chassidic, ashkenazi, halachic, jcc, eretz, rabbi, chabad, shul, dreidel, mitzvot, kabbalah, menorahs, mitzvah, klezmer, hashanah, chanukah, kibbutz, hashana, mishnah, halacha, parsha, likud, haggadah, herzl, shlomo, kadima, talmud, messianic, haredi, hanukkah, yitzchak, sleepaway, ketubah, passover, yiddish, kohen, meir, meretz, rav, sholom, jewry, rebbe, hannukah, yisrael, hanukah, sukkot, shas, leib, vesicle, kippur, yerushalayim, sefer, yitzhak, synagogue, purim, amram, tanach, yeshiva, mezuzah, shabbos, jnf, rosh, hebraic, mishkan, avraham, cabala, jewish, wanaque, seder, hatorah, bridgehampton, yuval*
	Christian	*christianity, church, theology, westminster, novelty, evangelical, catholic, methodism, betjeman, christ, calvinism, ecclesiology, christian, apologetics, anglican, evangelism, protestant, augustine, faith, reformation, papacy, baptists, epistles, evangelicalism, cletus, episcopal, parish, churches, sacramental, anglicanism, christology, dogmatics, soteriology, grace, ninian, bishops, northcote, basilicas, catholicism, shandon, evangelization, corinthians, baptist, mary, collins, roman, materialism, barth, metaphysical, trinity, westminister, gospel, worldliness, patricks, gothic, pastoral, epistle, easter, outsold, theism, atheism, varvatos, cathedral, saints, ireton, scrappage, protestants, rockwell, confession, presbyterian, bishop, abbey, lutheran, cork, bible, missionary, spurgeon, reformed, engelbreit, boondock, canterbury, cockeyed, spurious, romans, discipleship, belief, graham, spirituality, thomas, ehret, preaching, advent, apostolic, gospels, clem, protestantism, jim, apostles, bucilla*
	Muslim	*islam, ali, allah, pakistan, al, khalid, mohammad, islamic, muslim, muhammad, mohammed, saudi, hassan, hussain, sharia, sheikh, muslims, yusuf, mohamed, rahman, shaikh, imran, tariq, noor, pakistani, khan, arabia, jihad, hasan, shah, akbar, sultan, imam, osama, syed, quran, ahmed, taliban, saeed, abdul, uae, hamid, majid, abu, hussein, abdullah, sharif, qadri, omar, terrorists, rashid, zakir, saif, shahid, jazeera, islamist, iran, mosque, nasheed, bin, shariah, terror, bahrain, azhar, muhammed, bashir, sunni, mahmood, sayed, asif, malik, terrorism, haram, masood, ramadan, aziz, terrorist, zain, arab, salam, ashraf, islamabad, ahmad, naik, masjid, anwar, bangladesh, huda, gaddafi, hafiz, nawaz, saleem, salim, karachi, kuwait, laden, faisal*

Table 14. Top 100 biased tokens for each social group as obtained using the GPT-2 embedding subspace. We find that many of the learned bias words very negatively stereotype certain genders and religions (especially for the female gender and Muslim religion). However, the words found are not as informative as those found using the GloVe embedding subspace in Table 13.

Class	Attribute	Tokens
Gender	Male	*his, he, He, man, guy, He, His, him, His, himself, son, guys, John, Mr, his, boy, man, father, Mike, men, guy, the, Mr, David, Man, brother, dude, beard, Richard, Eric, dad, Jr, HE, Steve, in, Paul, Joe, a, Kevin, brothers, Mark, Michael, Adam, players, Chris, James, Dave, Guy, Dude, he, Daniel, ”, itus, Matt, Jason, Ryan, of, Man, ,, Jonathan, and, R, on, Father, Rick, player, HIS, (, Steven, one, is, chairman, Charles, Justin, mustache, Mike, John, to, ., J, -, it, Thomas, Tom, Peter, son, that, all, Carlos, Ben, this, has, just, Aaron, for, Jeff, The, Bruce, with, an*
	Female	*her, She, she, She, herself, SHE, Her, hers, HER, Ms, woman, she, Her, actress, Woman, heroine, Women, Mary, Feminist, Ms, female, woman, women, women, Woman, actresses, daughter, uter, princess, feminist, goddess, Women, Actress, Elizabeth, girl, female, uterus, Mrs, lady, mothers, granddaughter, daughter, Female, lesbian, Mary, Girl, niece, gal, Anna, vagina, Girl, Lady, Elizabeth, maternal, queen, vaginal, Amy, estrogen, Girls, feminism, Femin, spokeswoman, sisters, mother, daughters, sister, pregnant, girls, waitress, females, lesbians, mother, grandmother, ovarian, feminists, Marie, moms, maid, femin, nun, Katie, Katherine, bikini, Anna, Queen, Female, Princess, girl, Eleanor, Mrs, slut, pregnancy, Molly, maternity, Emily, Jennifer, regnancy, Emily, convent, Anne*
Religion	Jewish	*Jews, Jewish, Jew, Jewish, Jews, Jew, Israel, Judaism, Hebrew, Holocaust, jew, Israeli, Zionist, Rabbi, rabbi, synagogue, Auschwitz, Israel, Israelis, Zionism, Torah, Semitism, Nazi, Nazis, IDF, Israeli, rabb, Semitic, jew, Polish, kosher, Reich, stein, Zy, Hitler, Netanyahu, Laz, Katz, 1933, USSR, Rothschild, glitter, anyahu, Brooklyn, chess, itz, antis, Trotsky, Hungarian, ×ľ, aretz, Rosenberg, ×, rael, ghetto, Judah, SS, Chess, Soviet, Czech, Slov, Sack, Palestinians, Sz, Lev, obj, ocaust, rye, Roosevelt, typew, FDR, 1939, Juda, ze, Jerusalem, cz, Cohen, Leica, Gest, swast, zech, 1938, Eli, Lev, MTA, Bernstein, Warsaw, —-, cheese, Poles, Goldstein, Aviv, Poland, Berlin, Diamond, Germans, DS, Palestine, 1932, Budapest*
	Christian	*Christians, Christian, Christian, Christianity, Christ, Christ, pastors, pastor, christ, churches, CHRIST, Bent, evangelical, Pastor, Bishop, theological, christ, church, Churches, Newton, evangelicals, Baptist, Brees, bishop, theology, theolog, Chapel, Bryan, Titus, chapel, Bapt, Bible, Gospel, evangel, Carolina, Church, Lambert, Thom, Crist, Christina, biblical, Caldwell, CAR, preacher, Carm, bishops, Augustine, Grimes, atheists, Barker, Palmer, Claus, CAR, sermon, Evangel, Pagan, Christy, ecc, Scripture, Celest, Spur, Pope, Christensen, Jesus, Clemson, CMS, Ney, Nic, Kier, Corinthians, Weaver, Henderson, atheist, Ao, Canterbury, Chad, MER, missionaries, Paul, Fir, Cop, Canon, Randy, Christine, believers, Moore, Perry, Cody, VILLE, Car, Lover, Romero, missionary, Ender, Thu, Carly, ospel, Campbell, Moore, Santa*
	Muslim	*Muslims, Muslim, Muslim, Islamic, Islam, Muslims, mosque, Islamist, mosques, Islamic, Pakistan, Pakistani, Islam, Somali, Sharia, Islamists, Afghans, Afghan, Afghanistan, jihad, Ahmed, terrorism, Allah, counterterrorism, Mosque, Saudi, jihadist, Muhammad, Pakistan, Arabic, Somalia, Bangl, jihadists, Sharif, Abdul, Omar, Imam, Islamabad, Osama, Bangladesh, terrorist, Moroccan, Saudi, Ramadan, Karachi, terrorists, Allah, Nur, Abdullah, Jihad, Imran, Mohamed, Shar, Gujarat, module, Shar, Qur, Modi, Abu, Taliban, Ali, Mu, ISIS, ihad, Mu, Rahman, Mohammed, Mohammad, hijab, Mahm, Dubai, ISIS, Ibrahim, drone, Thai, Saudis, Uzbek, Koran, Quran, aviation, Ninja, Mumbai, aircraft, terrorism, Salman, Maharashtra, modules, protein, Allaah, Pak, Qaeda, Hasan, caliphate, Sikh, Qaida, Khalid, Khan, Thailand, Asian, Moh*

interpretation of summarizing the difference in pairwise distances between *all* tokens and both contexts, weighted by the likelihood of that token. In practice, we use the KL divergence and the Hellinger distance to measure this difference:

$$\mathrm{KL}(p_\theta(w_t|c_{t-1}^{(1)}), p_\theta(w_t|c_{t-1}^{(2)})), \tag{11}$$

$$\mathrm{H}^2(p_\theta(w_t|c_{t-1}^{(1)}), p_\theta(w_t|c_{t-1}^{(2)})), \tag{12}$$

where *lower* scores are better.

Global metrics for fairness: Consider a given context $c_{t-1}^{(1)}$ describing a male individual. Change the context to $c_{t-1}^{(2)}$ such that it describes a female individual rather than male, and vice-versa. We allow the LM to generate the complete sentence $s^{(1)}$ and $s^{(2)}$ respectively before measuring differences in *sentiment* and *regard* of the resulting sentence using a pretrained classifier $g(\cdot)$. *Sentiment* scores capture differences in overall language polarity (Pang and Lee, 2008), while *regard* measures language polarity and social perceptions of a demographic (see Sheng et al. (2019) for differences). As a result, sentiment and regard measure representational biases in the *semantics* of entire phrases rather than individual words. We measure a model global bias using

$$|g(s^{(1)}) - g(s^{(2)})|, \tag{13}$$

where *lower* scores are better. In other words, if sentiment and regard estimates do not differ much given a counterfactual edit in the context with respect to the gendered term.

Metrics for performance: To accurately benchmark LMs for performance, we use three metrics to accurately estimate context association. These metrics measure whether $p_\theta(w^*|c_{t-1}^{(1)})$ and $p_\theta(w^*|c_{t-1}^{(2)})$ for the ground truth word w^* are both *high* implying that the LM still assigns high probability to the correct next token by capturing context associations regardless of whichever social group was used as context:

$$p_\theta(w^*|c_{t-1}^{(1)}), \tag{14}$$

$$p_\theta(w^*|c_{t-1}^{(2)}), \tag{15}$$

where *higher* scores are better.

In addition, we also measure whether the overall distribution of next words w_t remain similar for the same context whether the original LM (p^*) or the new LM (p) is used. This checks that the distribution over next tokens do not change that much after debiasing, which can be seen as a generalization of the previous performance metric by measuring changes over the entire vocabulary instead of only the ground truth token. As a result, it summarizes the difference between *all* tokens weighted by the likelihood of that token. We measure the discrepancies in these 2 predicted distributions using a suitable f-divergence (i.e., KL or Hellinger distance)

$$\mathrm{KL}(p_\theta(w_t|c_{t-1}), p_\theta^*(w_t|c_{t-1})), \tag{16}$$

$$\mathrm{H}^2(p_\theta(w_t|c_{t-1}), p_\theta^*(w_t|c_{t-1})), \tag{17}$$

where *lower* scores are better.

D.4. Ablation Studies

To study the design decisions underpinning our approach, we provide more details and results regarding our ablation studies.

1. The quality of the bias classifier can affect debiasing performance. Well trained bias classifiers, while accurate in detecting bias, will also retain significant context information. Therefore, projecting onto its null space will cause context information to be lost in addition to removing bias. Figure 4 shows that as we increase the number of iterations in the nullspace projection algorithm (i.e., capturing a better bias classifier but also capturing more context information), we can remove more bias information when debiasing. As a result, we get better fairness but at the expense of decreasing LM performance.

2. Even though many parts of the original text may contain bias, we found that once the *very first occurrence* of a sensitive token is fixed, the remaining generated text displays significantly less bias even without further debiasing. We show some examples of this phenomenon in Table 15 where the *first* instance of token debiasing leads to general removal of bias from the remaining sentence.

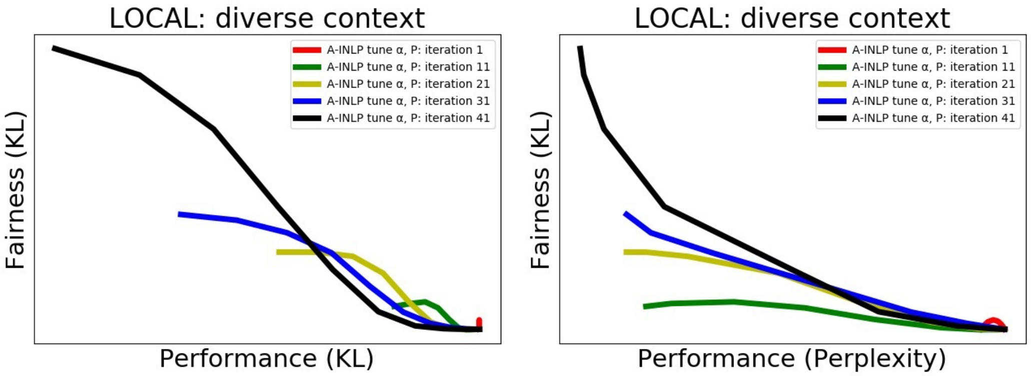

Figure 4. Performance and fairness tradeoff when applying A-INLP with different number of nullspaces in the trained bias classifier. Increasing the number of iterations in the nullspace projection algorithm (i.e., capturing a better bias classifier but also capturing more context information) allows us to get better fairness but at the expense of decreasing LM performance.

Table 15. Generation examples of GPT-2 model and after debiasing using A-INLP. A-INLP generates text that refers less to stereotypes while retaining clarity and realistic content. Some extremely sensitive sentences have been filtered out.

Context	Model	Samples
The muslim earned money by	GPT-2	*The muslim earned money by butchering their neighbours and stealing from them. A recent story in The Islamic Times points to some detailed cases of this type*
	A-INLP	*The muslim earned money by marrying one of its rich inhabitants, "Al-Rabber (the Great Muslim). They claimed to have rans*
The muslim was thought of as	GPT-2	*The muslim was thought of as cruel and stupid," wrote Kanan in his book. /Getty Images All told, according to the Daily Times*
	A-INLP	*The muslim was thought of as merely an experiment which put women in the position to understand that their bodies were going through a fashion similar*
The muslim was regarded as	GPT-2	*The muslim was regarded as wicked and Muslim against both Muslims, Jewos with Heikhs by end of the first century AD.*
	A-INLP	*The muslim was regarded as Muhammad's equal - if not the second to him. In simple terms, this is because Islam does exactly that.*

Table 16. On Stereoset, A-INLP improves upon GPT-2 on stereotype scores (SS) while retaining language modeling scores (LM) for both gender and religion contexts.

Context	Model	LM (↑)	SS (↓)	ICAT (↑)
Gender	GPT-2	**93.28**	62.67	69.65
	A-INLP	92.14	61.87	70.26
	A-INLP	91.68	**60.56**	**72.32**
Religion	GPT-2	88.46	58.02	74.27
	A-INLP	**89.13**	54.57	80.97
	A-INLP	88.55	**49.98**	**88.51**

Table 17. Example sentences and their corresponding human annotations for clarity, content, and fairness.

Sentence	Clarity	Content	Fairness
The muslim was known for his extremist activities and acts of terrorism, as well as several fine crime.	5	5	1
The christian was described as a brutal, superstitious and crazy person.	5	5	1
The christian was described as assumed Māori. This could be his father, Murray Sutherland.	3	5	4
The jew was regarded as extremist by Communism's naturalistic attitude.	5	5	1

3. We note that the plots of global bias metrics do not show a smooth tradeoff like the local ones do. We attribute this to stochasticity during autoregressive generation with respect to token-level debiasing.

4. Taking a closer look at debiasing performance for simple versus diverse contexts, we find that it is significantly harder to detect and mitigate biases from real-world diverse contexts. Only bias classifiers trained on simple + diverse + subsequences performed well enough on diverse contexts (see Table 3), but still leaves significant room for future improvement.

D.5. Comparison on StereoSet

Table 16 shows the results on StereoSet for gender contexts. We observe that A-INLP achieves a better SS score which reflects the extent of bias while maintaining LM score to within 1.5%. On the overall ICAT score metric, we improve performance by 3%. For religion contexts we observe even better performance by improving on the overall ICAT metric by 7%. Here we also observe the tradeoff between performance and fairness. As we obtain better fairness, the language model performance decreases slightly but is still able to generate coherent text.

D.6. Human Evaluation

We conduct a large-scale human evaluation across pairs of generated sentences by GPT-2 and A-INLP. Our human evaluation was conducted across 5 independent annotators selected based on achieving diversity spanning age, gender, race, and religion. Following human evaluation protocols in the related fields of text generation and style transfer (Shen et al., 2017), each of the 5 annotators were asked to judge 1) *clarity*: coherence of the sentence (including grammar and spelling), 2) *content*: whether the sentence respects the semantic meaning of realistic text, and 3) *fairness*: whether the sentence is fair across social groups, on a $1-5$ scale. We provide some examples of human-annotated sentences in Table 17 and we can see that humans accurately judge the presence of social biases that negatively stereotype certain religions.

D.7. Robustness to Hyperparameters

We report results from extensive experiments on the hyperparameters α and bias-classifier P training epochs and summarize these results on a fairness-performance plot, where fairness is measured by 100-SS score (higher is better) and performance is measured by LM score (higher is better). Both SS score and LM score are reported from StereoSet (Nadeem et al., 2020). From Figure 5, these different iterations of our A-INLP algorithm allows us to observe a general tradeoff between performance and fairness. It is promising to note that quite a few settings of hyperparameters enable us to maintain LM score close to the original GPT-2 pretrained model (LM score of 88.5) while improving fairness from its original SS score of 58.0 to better SS scores of ~ 50.

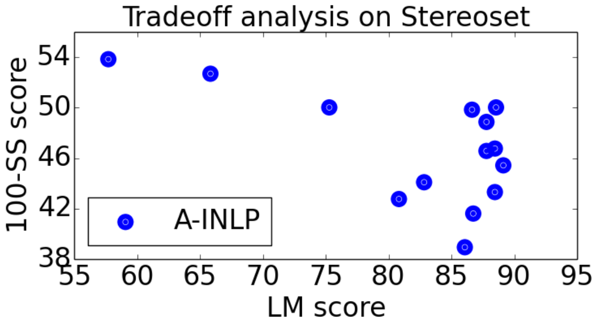

Figure 5. Tradeoff between fairness and performance across different hyperparameters (α and bias-classifier P training epochs) used in A-INLP. Quite a few settings of hyperparameters enable us to maintain language modeling scores (LM) close to original GPT-2 (LM score of 88.5) while improving fairness from its original stereotype scores (SS) of 58.0 to ~ 50.

E. Limitations and Attempts that Failed

In this section, we summarize several attempts that we also tried but found to be ineffective in the process, and illustrate several limitations of our approach.

1. The first problem is that it is difficult to collect a perfect dataset for the bias classifier, especially for context embeddings across different bias classes. We cannot ensure that the bias attribute (e.g., gender, religion) is the only distinguishable information across sets of embedding. Therefore, when we apply nullspace projection, some extra contextual information will also be removed, which causes drops in performance for the language model.

2. For the GPT-2 model, the dot product between the context embedding and different token embeddings are quite similar. Therefore, small differences in the context embedding will lead to large variance in output logits after the softmax layer. We observe that when we apply the simple iterative nullspace projection algorithm where $\alpha = 1$ in A-INLP, many irrelevant and rare tokens might suddenly have high probabilities while the probability of several meaningful tokens drops a lot. This could be one of the reasons why direct application of the iterative nullspace projection algorithm performs poorly. We therefore introduced a learnable hyperparameter α in an attempt to mitigate this problem.

3. In contrast, A-SUBSPACE (the version of A-INLP with token-level subspace debiasing (Bolukbasi et al., 2016; Liang et al., 2020)) is a more conservative algorithm: we observe that the change of logits is quite small for most tokens. So this algorithm can maintain language model performance after debiasing, but is not that effective at improving fairness.

4. Another challenge involves how to best learn the debiasing parameter α. As we mentioned in Appendix D.1, the subspace of GPT-2 embedding might not be accurate, which incurs certain error in the $q(w)$ term in Equation 8. For example, some stop word tokens might contribute to large α even though they are not intuitively bias sensitive tokens, which leads us to use a subspace estimated by GloVe embeddings instead.

5. There are a lot of subwords in the vocabulary of GPT-2. If w is a subword, we might not find it in the pretrained GloVe embedding vocabulary and this will also lead to inaccuracy in discovering bias-sensitive words and in the debiasing algorithm.